
TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: B2646 – Informační technologie

Studijní obor: 1802R007 – Informační technologie

**Automatické generování
zkouškových testů**

**Program for automatic final
exam tests generation**

Bakalářská práce

Autor:	Jakub Ponikelský
Vedoucí práce:	Ing. Tomáš Martinec, Ph.D.
Konzultant:	Ing. Martin Vlasák

V Liberci 1. 4. 2011

Prohlášení

Byl(a) jsem seznámen(a) s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval(a) samostatně s použitím uvedené literatury a na základě konzultací s vedoucím bakalářské práce a konzultantem.

Datum:

.....

Podpis:

.....

Jakub Ponikelský

Poděkování

Touto cestou bych především rád poděkoval vedoucímu své bakalářské práce Ing. Tomáši Martincovi, Ph.D. za jeho podporu a mnoho cenných rad při vytváření bakalářské práce, dále také Ing. Martinu Vlasákovi za jeho cenné připomínky a rady při konzultacích.

Ještě bych chtěl poděkovat prof. Ing. Zdeňku Plívovi, Ph.D. za jeho typy a cenné informace, které nám předal během předmětu Bakalářský seminář.

Za podporu při studiu a tvorbu potřebného zázemí velmi děkuji své rodině, která mi vždy byla velkou oporou.

Abstrakt

Cílem této bakalářské práce je navrhnout a vytvořit program pro generování zkouškových testů. Teoretická část práce je zaměřena na již existující programy, které umožňují vytvářet a zveřejňovat testy. Jsou zde popsány jejich vlastnosti a zhodnocena jejich vhodnost pro zadaný problém. Dále je v této části popsáno vývojové prostředí, které bylo zvoleno pro tvorbu vlastního řešení.

V praktické části je navrženo a vytvořeno vlastní řešení, umožňující z dodané databáze otázek a odpovědí generovat testy s požadovanými parametry. Výstupem z programu je tisková sestava, obsahující zvolené množství testů v uživatelem zvoleném formátu. Program také vytváří databázi použitých testů, aby bylo možné porovnat, jak se aktuální test liší od testů minulých. Na závěr je program otestován na předmětu ze základního kurzu bakalářského studia.

Klíčová slova:

sestavení testu,
generování otázek,
tisk testu,
databáze testů,
vytvoření programu,
existující programy

Abstract

The objective of this thesis is to design and create a program for exam tests generation. The theoretical part is aimed at existing programs that allow you to create and publish tests. Here are described their properties and evaluated their suitability for a given problem. In this section of thesis is described development environment, which was selected for the creation of the program.

The practical part is to design and create a program that generates a test with the required parameters using a database of questions and answers. The output of the program is printed pages that contain the selected number of tests in a user-selected format. The program also creates a database of previously generated tests. It is used to compare how the current test differs from the previously generated tests. Finally, the program printed a sample test from the basic course of undergraduate study.

Keywords:

build test,
generating questions,
print test,
database of tests,
program development,
existing programs

Obsah

Prohlášení.....	3
Poděkování.....	4
Abstrakt.....	5
Abstract.....	6
Obsah	7
Seznam obrázků.....	9
Seznam zkratk	10
1 Úvod.....	11
2 Existující programy pro tvorbu testů	13
2.1 Generátor jedinečných zadání 1.4 (GEJZA 1.4)	13
2.2 QuizFaber 2.12.2	16
2.3 Test Generator	18
2.4 Test 1.1.4.25	20
2.5 Quiz Builder 2.0.0	22
2.6 DoTest 4.55	24
3 Cíl práce: požadovaná funkčnost programu	28
4 Nástroje použité při tvorbě vlastního řešení	30
4.1 Vývojové prostředí wxDev-C++ 7.3.1	30
4.2 Databázový systém MySQL 5.5	31
4.2.1 Použití MySQL v prostředí wxDev-C++	32
4.3 Knihovna wxWidgets	32
5 Program pro Generování Testů (vlastní řešení)	34
5.1 Použité nástroje	34
5.1.1 Nástroje wxDC	34
5.1.2 Eventy (události).....	35
5.1.3 Zmíněné datové typy	36

5.1.4	Další pojmy	37
5.2	GUI pro nastavení parametrů testu	37
5.2.1	Třída „souborOtazek“	37
5.2.2	Třída „modulOtazky“	38
5.2.3	Dialogové okno „dlgVyberOtazek“	38
5.3	Práce s databází testů	41
5.3.1	Třída „praceMySQL“	41
5.3.2	Dialogové okno „DbNamePassDlg“	43
5.4	Vygenerování testu.....	43
5.4.1	Třída „popisProGenOtazky“	43
5.4.2	Třída „MyThread“	45
5.5	Nástroje pro vytvoření tiskové sestavy	46
5.5.1	Třída „MyPanel“	46
5.5.2	Třída „MyRower“	47
5.5.3	Dialogové okno „VyberHlavickyDlg“	48
5.5.4	Dialogové okno „VyberTestuDlg“	49
5.5.5	Třída „MyPrintout“	49
5.6	Doplňující funkce.....	50
5.6.1	Dialogové okno „Porovnanitestudlg“	50
5.6.2	Dialogové okno „vytvorSouborTestuDlg“	51
5.7	Hlavní okno aplikace „PpGTfrm“.....	51
6	Závěr	56
	Seznam použité literatury	57
	Příloha A – Obsah CD	59
	Příloha B – Ukázka testu č. 1.....	60
	Příloha C – Ukázka testu č. 2.....	61

Seznam obrázků

Obrázek 2.1: Hlavní menu programu Generátor jedinečných zadání.....	15
Obrázek 2.2: Titulní obrazovka programu QuizFaber.....	16
Obrázek 2.3: Úvodní menu programu Test Generator	18
Obrázek 2.4: Menu programu Test	20
Obrázek 2.5: Úvodní obrazovka programu Quiz Builder.....	22
Obrázek 2.6: Hlavní menu programu DoTest.....	25
Obrázek 5.1: Grafická podoba okna „dlgVyberOtazek“	39
Obrázek 5.2: Schéma použité databáze	42
Obrázek 5.3: Dialogové okno „DbNamePassDlg“	43
Obrázek 5.4: Dialogové okno pro vybrání hlavičky testu	48
Obrázek 5.5: Dialogové okno pro výběr vzhledu testu	49
Obrázek 5.6: Ukázka porovnání testů.....	50
Obrázek 5.7: Jednoduchý editor souboru otázek	51
Obrázek 5.8: Hlavní okno aplikace při výběru parametrů hlavičky	54

Seznam zkratek

SW = Software – programové vybavení počítače

GEJZA – Generátor jedinečných zadání

DOS = Disk Operating System – diskový operační systém (starší OS)

ČVUT – České vysoké učení technické v Praze

HTML = HyperText Markup Language – značkovací jazyk

PC = Personal Computer – osobní počítač

PDF = Portable Document Format – formát pro ukládání dokumentů

USB = Universal Serial Bus – sériová sběrnice

DPH – daň z přidané hodnoty

IDE = Integrated development environment – Integrované vývojové prostředí

XML = Extensible Markup Language – značkovací jazyk

XRC = XML Resource – značkovací jazyk

GDB = GNU project debugger – druh ladícího nástroje

SQL = Structured Query Language – strukturovaný dotazovací jazyk (práce s databází)

GPL = GNU General Public License – druh licence

API = Application Programming Interface – seznam funkcí a procedur určité knihovny

GUI = Graphical user interface – grafické uživatelské rozhraní (okna pro ovládání)

PpGT – Program pro generování testů (program vytvořený v této práci)

ID – identifikace (například jednoznačná identifikace řádku v SQL tabulce)

1 Úvod

Zkouškový systém se stal základem naší, ale i zahraniční společnosti. Je na něm založeno naše školství, ať už se jedná o maturity, statní závěrečné zkoušky na vyšších stupních vzdělávání nebo jakékoliv osvědčení. Příkladem zmíněných osvědčení mohou být certifikáty nabízené v oboru informačních technologií od předních zahraničních společností, jako je Microsoft Corporation, Cisco Systems, Google a mnoho dalších. Zkrátka, pokud chce člověk ve svém oboru nabídnout něco víc, zkouškám se nevyhne.

Tento systém vznikl již ve starověké Číně, kde se pomocí úřednických zkoušek vybírali správci některých částí země. Systém se postupem let vyvíjel, standardizoval a později se jím inspirovaly evropské národy. V současnosti jsou nejčastějšími druhy zkoušek, zkoušky ústní, písemné, praktické nebo fyzické.

U písemných zkoušek většinou závisí na tvůrci, jaký formát písemné práce si zvolí. Jestliže se ovšem jedná o nějakou významnou zkoušku je formát práce předepsán zákonem či jiným standardem. Vývojem v této oblasti se některé styly otázek v testech staly běžnými, zejména jsou to otázky založené na výběru z několika různých odpovědí nebo otázky, na které musí zkoušený odpovědět pomocí několika vět. Záleží na zkoušejícím, jestli zvolí jednotný styl pro celý test nebo bude možnosti kombinovat.

V testech s předepsanými odpověďmi musí zkoušený vybrat všechny odpovědi, které jsou správné. Podle tohoto kritéria se testy dělí na testy s jednou správnou odpovědí a na testy s proměnným množstvím správných odpovědí. Tento druh testu má několik výhod i nevýhod. Výhody je možné spatřit v tom, že zabírají méně času na zkoušení, lze jimi pokrýt široký rozsah témat a umožňují vytvořit více obtížnostních stupňů. Pomocí těchto testů se také lépe poskytuje zpětná vazba zkoušenému, proč takhle odpověď dobře je a takhle už není.

Bohužel tento formát testu má několik nevýhod, mezi ně patří to, že neumožňuje zkoušenému projevit větší znalost určité oblasti. Také je větší šance na úspěch plynoucí z toho, že se může zkoušenému povést i bez požadované znalosti v dané oblasti správnou odpověď odhadnout. Tento problém je zejména u testů s jednou správnou odpovědí. Problémem také bývá nepochopení termínu v otázce a z toho plynoucí nevědomost zkoušeného. Ovšem největší nevýhodou tohoto druhu testů je náročnost vytvoření takového testu, která je omezující zejména pokud by se měl test často opakovat, protože je pak nutné vytvářet mnoho různých verzí.

S nástupem počítačové techniky si člověk začal složité úkoly zjednodušovat. A to je i motivací vzniku této práce, navrhnout a vytvořit nástroj, který bude vhodný pro tvorbu testů s předdefinovanými odpověďmi. Kromě generování takového testu, bude poskytovat i další funkcionalitu pro pohodlí tvůrce.

Samozřejmě nějaká řešení tohoto problému již existují, to bude v této práci také zohledněno a na některá existující řešení se zaměří podrobněji s cílem ověřit jejich použitelnost pro tvorbu testů v základním kurzu bakalářského studia.

2 Existující programy pro tvorbu testů

Na internetu, ale i na trhu s počítačovým softwarem (SW) se pohybuje velké množství programů, které umožňují generování a publikování testů. Nejrozšířenější jsou pravděpodobně programy, vytvořené pro účel ověření získaných vědomostí, které jsou přidávány k různým výukovým SW. Vhodným příkladem, může být generátor cvičných testů pro získání řidičského oprávnění, dostupný na webových stránkách Ministerstva dopravy (<http://etesty.mdcz.cz/>).

Dále je možné tyto programy najít v různých počítačových hrách, jako například adaptace známého pořadu *Who Wants to Be a Millionaire* u nás známého jako *Chcete být milionářem?* (existuje několik různých verzí této hry a to v angličtině i v češtině, verze v angličtině např. www.super-hry.com/online-349/ostatni/chcete-byt-milionarem). Dalším zástupcem jsou například IQ testy, kterých je možno na internetu také najít velké množství (např. <http://www.celysvet.cz/iq-test-zdarma-i-s-vysledky>).

Všechny zmíněné programy se vyznačují tím, že není možné měnit databázi otázek. Také nelze porovnávat minulé testy s tím současným a ve většině případů není možné ani vytisknutí vygenerovaného testu. Tudíž jsou pro generování zkouškových testů nepoužitelné. V tomto směru je nutné, zaměřit se přímo na SW pro generování testů. Ty se na internetu nevyskytují v takové míře jako výše zmíněné programy, ale několik zástupců je možné najít. Některé lze získat bezplatně, některé jsou už na profesionálnější úrovni a je nutné zakoupit si licenci. Několik takových příkladů bude popsáno dále.

2.1 Generátor jedinečných zadání 1.4 (GEJZA 1.4)

První nalezený a vyzkoušený program pro generování testů je GEJZA 1.4, jehož autorem je doc. Ing. Luboš Svoboda. Tento program vznikl v roce 2005 pro katedru stavebních hmot na ČVUT v Praze. GEJZA 1.4 využívá platformu DOS a je málo hardwarově náročný. Je vyvinut a odladěn pro používání v prostředí Windows XP, ale jeho stavba umožňuje práci i na systémech starších. [1] Pro fungování na systémech novějších je doplněn programem DOSBox. Slouží zejména pro vytváření testů v podobě množiny otázek, na které účastník testu okamžitě písemně odpoví. Dále jsou popsány jeho vlastnosti a funkce.

Program si po spuštění ve svém adresáři automaticky vytváří několik složek a textových souborů, které jsou nutné pro jeho chod. Jedná se o složky pojmenované

01BLOK–99BLOK (automaticky se vytvoří prvních deset složek), ve kterých program očekává báze otázek pro generování testu. Dále složka STATUS, ve které se nachází nastavení programu. Nakonec složka TESTY, do které se jak již název napovídá, vytváří vygenerované testy.

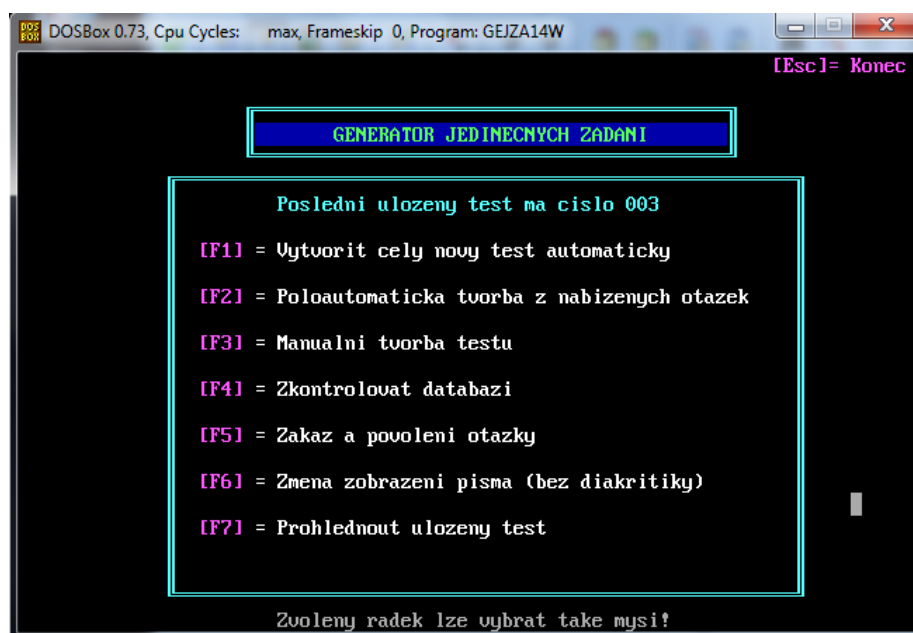
Ve složkách 01BLOK až 99BLOK jsou báze otázek a ty jsou uloženy v datových souborech v podobě obvyčejného textu. V textu těchto souborů se pak pravidelně opakují tyto položky (každá na samostatném řádku): šestimístné číslo otázky (100000–999999), číslo představující obtížnost otázky (1–9), otázka a odpověď. Otázka i odpověď mohou být víceřádkové, ale musí být vloženy do uvozovek. [1]

Úprava nastavení se provádí pomocí šesti základních souborů v adresáři STATUS. Díky těmto souborům je možno měnit jak nastavení estetické tak funkční. Estetické nastavení je v tomto případě změna ikony programu. Co se týče funkčního nastavení, je tu soubor uchovávající seznam použitých otázek (bohužel uchovává se jen číslo otázky, takže při změně vstupních souborů může dojít k potlačení nových otázek), soubor obsahující zakázané otázky (opět pouze pomocí čísel otázek), soubor obsahující použitou znakovou sadu, konfigurační soubor a soubor obsahující poznámku k testu. Konfigurační soubor ovlivňuje například počet otázek v testu (až 99), počet otázek s určitou obtížností v testu, promíchání otázek atd.

Pro sestavení testů jsou v programu k dispozici tři různé módy: automatický, poloautomatický a manuální. V automatickém módu se výběr otázek do testu provádí náhodně. Náhodně se zvolí báze otázek (pokud je dostatek bází, tedy pokud není některá báze prázdná, je pokaždé vybrána jiná), z té se poté náhodně vybere soubor s otázkami a z něj je náhodně vybrána vhodná otázka (podle obtížnosti). Aby se test co nejvíce lišil od minulých, je výběr otázky upraven. U každé otázky se počítá, kolikrát byla již vybrána do předchozích testů a to pomocí zmíněného souboru ve složce STATUS. Přednost dostává otázka nepoužitá, respektive nejméně používaná. Pokud je již každá otázka z báze použita stokrát, program nahlásí vyčerpání báze. [1]

V poloautomatickém režimu je postup stejný, jen se otázka nezařadí ihned do testu, ale program se zeptá, zda má otázku do testu zařadit. V manuálním režimu si otázky vybírá sám uživatel s tím, že program mu poskytuje údaj o tom, kolikrát již byla daná otázka použita.

Obsluhování programu je velmi jednoduché. Z nabídky programu si uživatel vybírá funkčními klávesami, klávesami na numerické klávesnici nebo myší. Tisk tímto programem vygenerovaných testů je realizován pomocí prostředků Windows, konkrétně programu Poznámkový blok (Notepad), do něhož se testy převádějí. Při stisku klávesy F8 v hlavním menu by se podle autora měla objevit podpora tisku. Bohužel ve verzi pro Windows 7 tato možnost chybí, takže tisk lze provádět opravdu pouze pomocí poznámkového bloku. [1]



Obrázek 2.1: Hlavní menu programu Generátor jedinečných zadání

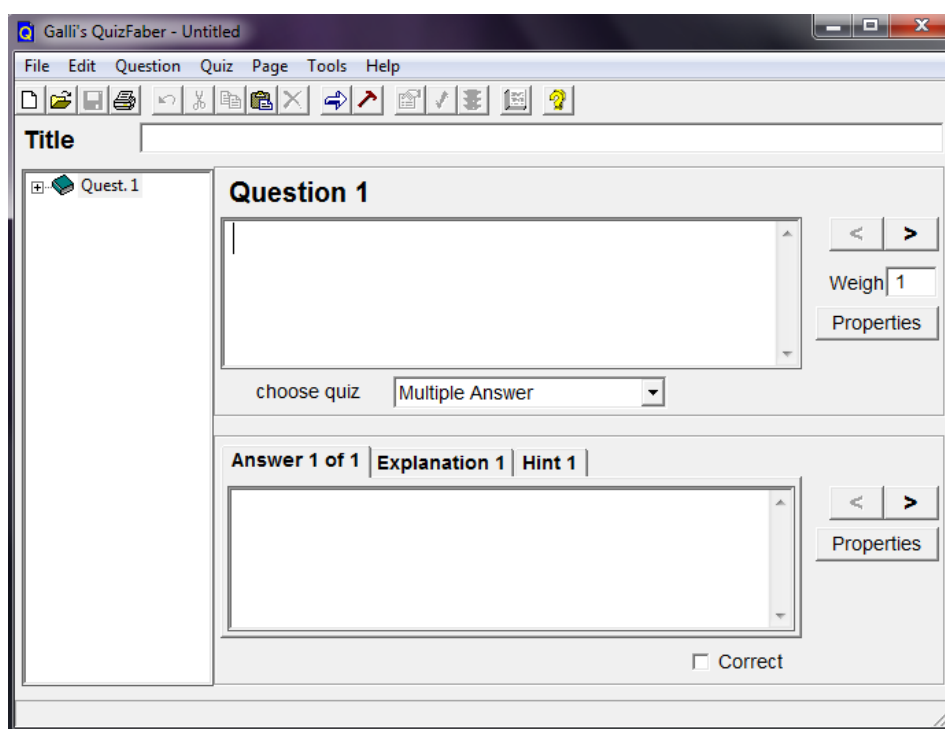
Testy se ukládají ve třech různých verzích. Jedna je textový soubor obsahující pouze otázky, druhá je textový soubor, který obsahuje odpovědi a třetí je soubor formátu HTML, ve kterém jsou otázky i odpovědi spolu s dalšími informacemi, jako je obtížnost otázky, číslo otázky v databázi. HTML soubor je částečně zformátovaný. Pojem „částečně zformátovaný“ znamená, že jsou otázky zvýrazněny tučným písmem.

Závěrečné zhodnocení: program GEJZA je užitečný na výrobu krátkých testů „otázka-odpověď“. Nicméně uchovávání statistik pouze pomocí čísel otázek vyžaduje při změně jen jedné báze otázek kontrolu, zda nová otázka nemá číslo staré nebo úplné smazání statistiky o výběru otázek. Neumožňuje vytvoření testů „abcd“. Prostředí DOS už dnes také není nikterak uživatelsky přívětivé. Za ocenění stojí jeho postup výběru otázek a jeho možnost použití i na starších počítačích.

2.2 QuizFaber 2.12.2

Další program, který byl vyzkoušen je QuizFaber. Jak již název napovídá, nejedná se o český program. Jeho autorem je Ital Luca Galli. Vznikl již v roce 1997, ale autorem je vylepšován, až do současné doby. Je to freeware určený pro práci v systému Windows. Podporuje několik světových jazyků, bohužel čeština podporována není a to přináší nepříjemnosti s diakritikou. Jeho primární funkcí je tvorba HTML testů. Bez znalosti jazyka HTML či JavaScriptu ale umožňuje i tvorbu testu a jeho následný tisk. Dále jsou popsány jeho základní vlastnosti a funkce. [2]

Po spuštění programu se ihned objeví pole pro titulek testu, velké textové pole a další možnosti nastavení jak pro otázku, tak pro odpověď (viz Obrázek 2.2). Zde lze nastavit váhu otázky a také text, který se zobrazí jako nápověda k otázce či text, který se zobrazí po zodpovězení otázky. V položce „Properties“ lze k otázce/odpovědi přiřadit zvuk, video, obrázek, či Java applet.



Obrázek 2.2: Titulní obrazovka programu QuizFaber

Tento program podporuje několik druhů otázek: otázku typu „abcd“ s jednou nebo více správnými odpověďmi, text doplněný podotázkami, na které lze odpovědět pouze ano či ne, otázka na kterou musí uživatel písemně odpovědět, přiřazování slov ze dvou množin nebo doplňování chybějících slov ve větě.

Co se funkcí týče, je samozřejmostí možnost importovat nebo exportovat jakýkoli test a volba písma. V programu lze nastavit hlavičku testu (autor, organizace, datum a název testu), nějaké informace zobrazené na začátku testu a také závěrečný text. Také existuje možnost z námi vytvořených otázek pro právě generovaný test některou vynechat. V nastavení testu lze nalézt velké množství voleb, jako například časový limit, maximální a minimální hodnocení v závislosti na správných odpovědích, zamíchání otázek, vložení tlačítka pro předčasný konec testu, různé zvuky pro správnou/špatnou odpověď a mnoho dalších možností. Také lze vytvořit verzi testu pro mobilní telefon, nicméně ta nepodporuje určité druhy otázek. Také zde nalezneme rozsáhlou nápovědu, bohužel pouze v angličtině.

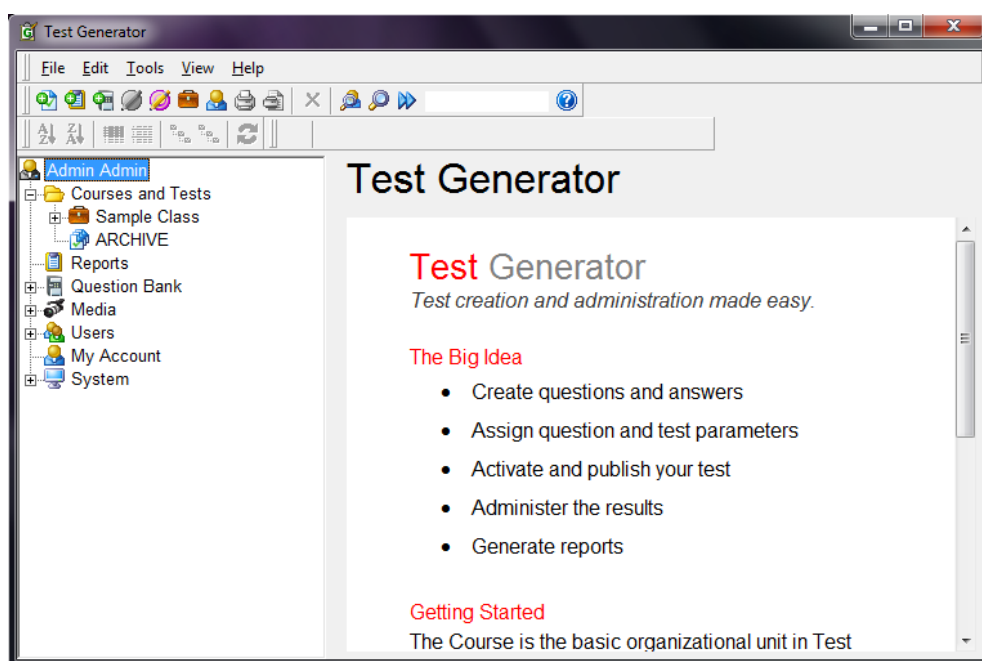
Možnosti pro vytváření HTML testů jsou opravdu rozsáhlé, například velké množství vzhledů testu. Dokonce je tu možnost (po nainstalování web serveru a dalších doplňků, které autor uvádí na svých webových stránkách) test nastavit tak, aby se spustil pouze po zadání platného jména a hesla a uchovával výsledky. [2] Bohužel s podporou tisku testů je to o něco horší. Sice zde je možnost vytisknout test, jak s označenými správnými odpověďmi, tak čistě neoznačený test, ale například vložené obrázky u otázek na vytištěném testu nejsou. Také zmíněné nastavení testů pro tištěnou formu neplatí, například po nastavení zamíchání otázek tištěný test bude pořad v původním pořadí. Další nedostatek tištěné verze je například ten, že v HTML verzi jde k otázkám, kde se doplňuje text do věty přiřadit nápovědu, ale ta se zobrazí pouze ve vyplněné tištěné verzi.

Závěrečná rekapitulace: program QuizFaber je velmi komplexní nástroj pro generování HTML testů, podporující i tisk. Jeho výhodami jsou: možnost vytisknout, jak „opravenou“ verzi testu, tak verzi „prázdnou“, lepší formátování tištěných verzí oproti předchozímu programu (včetně větší možnosti upravit hlavičku testu) a možnost vytvořit test typu „abcd“ s jednou či několika správnými možnostmi. Nevýhodami, které ovšem z velké části plynou z toho, že program je zaměřen právě na HTML testy, jsou: neúplná podpora tisku (některé funkce, které v HTML verzi najdeme, u tištěné verze nenajdeme), není zde možnost nahrát několik databází otázek, ze kterých by program do testu vybral náhodné z nich a také to, že program nepodporuje diakritiku.

2.3 Test Generator

Třetí program, který byl v rámci práce vyzkoušen je Test Generator. Jeho vydavatelem je americká firma, se sídlem v Atlantě, Fain & Company. První verze vyšla před rokem 1999 (podle internetových stránek autora je první vydané vylepšení datováno na prosinec 1999). Test Generator je zástupce komerčních programů pro generování testů. Jedná se o poněkud komplexnější program, který umožňuje vytvářet skupiny uživatelů, kteří poté podle svých práv mohou testy vytvářet nebo je pouze absolvovat. Na internetových stránkách autora je možné po vyplnění kontaktních údajů stáhnout trial (30denní) verzi. Díky této verzi je možno popsat základní funkce. [3]

Po nainstalování programu se v nabídce start objeví zástupci dvou spustitelných programů. Jedná se o Test Generator a Tester. Program Tester slouží k absolvování testů. Po jeho spuštění uživatel vyplní přihlašovací údaje a program oznámí, jestli je nutné nějaký test absolvovat a případně dá na výběr, který test chce uživatel absolvovat. Po spuštění programu Test Generator se také musí vyplnit přihlašovací údaje (při prvním spuštění je přihlašovací jméno „Admin“ a heslo také „Admin“). Po přihlášení je umožněn vstup do aplikace. Vlevo se nachází menu, pomocí něhož se program ovládá.



Obrázek 2.3: Úvodní menu programu Test Generator

V položce „System“ se nalézá historie přístupů (kdo a kdy sem vkládal test, absolvoval test, přidal uživatele aj.), zámky a základní nastavení programu. V nastavení lze po zakoupení plné verze vyplnit jméno a heslo pro aktivaci programu, je možné zde

měnit obrázek a zobrazovaný text při přihlašování do programu nebo nastavit výchozí formát při tisku testů (ten lze poté u libovolného testu měnit). V položce „My Account“ se mění heslo a údaje právě přihlášené osoby (může to být pouze „učitel“ nebo administrátor). Položka „Users“ slouží pro vkládání uživatelů. Jako administrátor lze přidávat „žáky“, „učitele“ nebo další administrátory. Pokud se přihlásí „učitel“ může vkládat pouze „žáky“. Uživatele je také možno třídit do různých skupin. O uživateli lze vyplnit několik údajů, ale povinné jsou pouze přihlašovací údaje.

V položce „Media“ jsou shromažďovány obrázky, zvuky, videa a další doplňky, které lze přikládat k otázkám. Položka „Question Bank“, jak již název napovídá, slouží k ukládání otázek. Tyto otázky je možno třídit do složek a podsložek. Takto uložené otázky je poté možné importovat do testu, který se později vytvoří. Program podporuje 11 variant otázek a to: otázka typu „abcd“ s jednou nebo více správnými odpověďmi, otázka ano/ne, doplňování slov do věty, otázka kde se musí odpověď vypsát, otázka kde se musí seřadit odpovědi, numerická otázka, přiřazování, otázka kde jsou odpovědi různě bodově ohodnoceny (člověk absolvující test se snaží vybrat nejlepší možnost), takzvanou esej a označovací otázku. Odpověď na esej nevyhodnocuje program, ale sám zadavatel. Označovací otázka vypadá tak, že je k ní přiřazen obrázek a v něm označena oblast, kam musí „žák“ kliknout. U každé otázky je možné nastavit její bodové ohodnocení a také to, jestli body budou jen za úplnou odpověď nebo stačí částečná odpověď, také lze za označení špatné odpovědi body strhávat atp.

V „Reports“ se automaticky ukládají různé statistiky o testech, úspěšnosti apod. A nakonec položka „Courses and Tests“. Zde se vytvoří skupiny a v těchto skupinách se vytvoří testy. Každé skupině testů lze přiřadit seznam uživatelů, kteří do ní patří a také seznam uživatelů, kteří jí vyučují. Když se ve skupině vytvoří test, je nutné vybrat z těchto uživatelů ty, kteří ho mají absolvovat.

Testy se vytváří označením skupiny a kliknutím na ikonu s názvem „New test“. Objeví se editor, ve kterém se vytváří otázky nebo lze otázky importovat z banky či souboru. Dále je zde možné ovlivnit nastavení testu, jako například písmo, jméno testu, časový interval pro splnění testu, časový limit na test, kolik procent je nutné na úspěšné splnění testu a další. Také je zde nastavení šablony pro tisk testu a samotný tisk testu.

Na závěr jsou v práci opět zhodnoceny klady a zápory programu. Kladem je komplexnost programu, vytvoření skupin uživatelů a jejich přiřazování k testům. Navíc

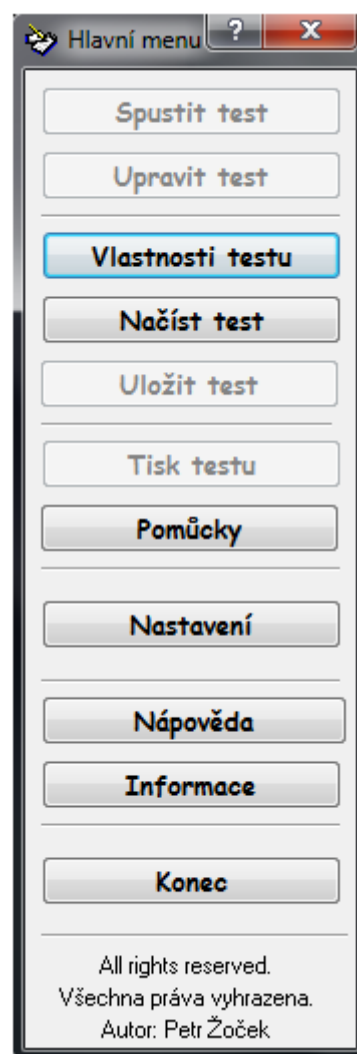
lze také test absolvovat několika způsoby: buď jako tištěný nebo přímo v programu na počítači. Lze zakoupit i verzi, kde se testy absolvují přímo prostřednictvím internetu.

Jako nevýhodu lze označit to, že u testu nelze nastavit náhodný výběr otázky z databáze a také slabší formátování tištěných verzí. Uživatel si sice může vytvořit šablonu a může si vytisknout, jak vyplněnou, tak nevyplněnou verzi. Když je test delší, nebere program ohledy na to, jestli je otázka na jedné straně či rozdělena na dvě. Další nevýhodou je zpoplatnění programu (10. 10. 2010 byla cena 350 dolarů [3]). Navíc, pokud by chtěl uživatel využívat testy po internetu, je nutno kontaktovat autory a tudíž se dá předpokládat další zpoplatnění této služby.

2.4 Test 1.1.4.25

Další vyzkoušený program se symbolicky jmenuje Test 1.1.4.25. Autorem je Petr Žoček ml. Jeho první verze byly shareware, vznikly v roce 1999 a byly napsány jazykem Visual Basic. Autor na programu dále pracoval, některé další verze již byly napsány v jazyce Delphi a autor umožnil přispěvatelům na svých internetových stránkách se podílet při vylepšování. Od roku 2003 je program freeware a zatím poslední verze byla vydána v roce 2004. Program je určen pro generování testů typu „abcd“, umožňuje testy jak spustit a vyhodnotit na PC, tak také vytisknout. V programu je podpora tří jazyků: čeština slovenština a angličtina. [4]

Prvním krokem k vytvoření testu je otevření menu „Vlastnosti testu“. Zde se nachází nastavení některých údajů: jméno testu a počet odpovědí, které se testovanému zobrazí nebo vytisknou. Test lze také zakódovat, aby nebyl čitelný bez správného klíče, či vybrat způsob hodnocení: chybové, bodové nebo procentuální. Je tu možnost nastavit i časový limit na otázku/test.



Obrázek 2.4: Menu programu Test

Poté je již možno vkládat otázky. Při otevření menu „Upravit test“ se zobrazí formulář, do kterého se vyplňují otázky a ke každé otázce je navíc možno vložit obrázek. Bohužel žádná položka není povinná, takže je možno vytvořit test, ve kterém není otázka ale jen odpovědi či naopak. Otázky je také možné vkládat v textovém editoru. Soubory jsou sice datové, ale jejich obsah je prostý text v určitém formátu, takže není problém soubor otevřít v textovém editoru a otázku tam přidat. Pokud editujeme soubor, tak na začátku je řetězec „CodN0020“, za ním následují čísla a znaky představující nastavení programu a po nich již otázky. Otázky jsou v tomto formátu (každá část na vlastním řádku): text otázky, číslo 1, text první odpovědi, číslo 0 nebo –1 (–1 představuje správnou odpověď) a takto následuje libovolný počet odpovědí.

Test je možné spustit s libovolným počtem otázek, které se z databáze vyberou náhodně a lze i zvolit náhodné promíchání odpovědí, nebo je test možno vytisknout s požadovaným nastavením jako jsou okraje, druh písma jednotlivých textů (nadpis, otázky, odpovědi aj.). Navíc je možné navolit si počet vytisknutých kopií a také si zvolit zda se nám poté vytisknou i správné odpovědi a to v libovolném počtu verzí či souhrnné výsledky všech testů (např. vytiskneme 10× pět různých testů + výsledky všech testů).

Již v úvodu byla zmíněna podpora více jazyků a tu lze ještě doplnit o libovolnou řeč. Stačí v nabídce Start ve Windows otevřít program „Překladač programu“, kliknout na volbu „Načít“, zobrazí se aktuální seznam jazyků a fráze v programu používané. Do volného pole je možné napsat název jazyku, který chceme do programu přidat a vyplnit všechny fráze v tomto jazyku. Po spuštění programu Test bude přidaná řeč v položce „Nastavení“ a bude možné ji aktivovat. Po instalaci je také k dispozici konvertor otázek ze starších verzí programu.

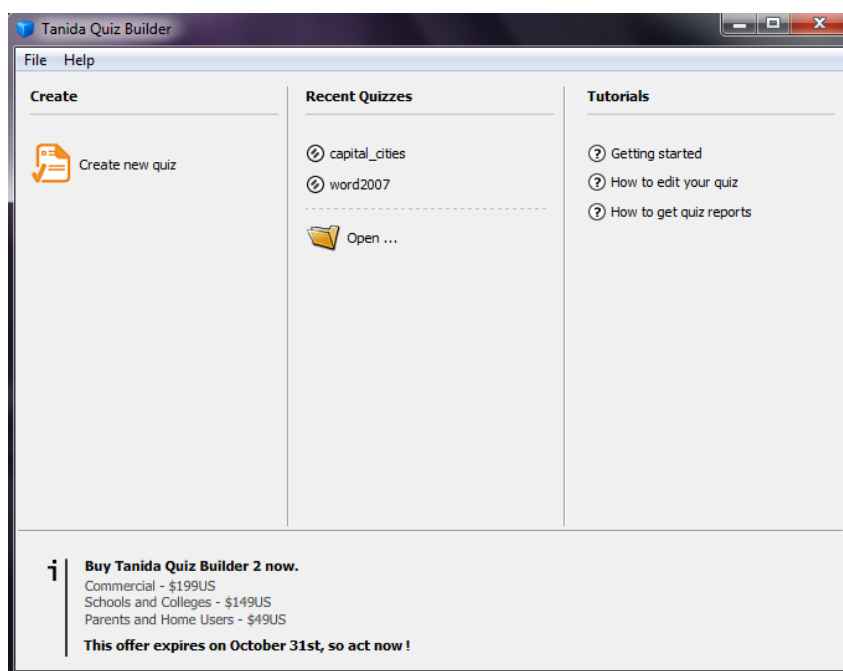
Zhodnocení kladů a záporů: ke kladům jistě patří základní formátování tištěných testů (ale není zde možnost vytisknout test například do dvou sloupců). Dalším kladem je náhodný výběr otázek z databáze a promíchání odpovědí. Také se program dá označit za vícejazyčný. Doplnit libovolnou řeč je, díky frázím v angličtině, otázka asi deseti minut.

Nevýhodou je možnost vložit neúplnou otázku a také to, že se při nedostatečném počtu otázek doplní test prázdnými otázkami. Také vybírání otázek z několika možných databází zde schází a statistika předchozích testů zde také není.

2.5 Quiz Builder 2.0.0

Další z vyzkoušených programů je opět zástupce komerčních SW pro tvorbu testů, jedná se o Quiz Builder 2.0.0. Jeho vydavatelem je firma Tanida Inc. Již podle jména vydavatele je patrné, že se nejedná o českou firmu, nýbrž o firmu se sídlem v Kanadě, konkrétně v Torontu. Verze 1.0 (první vydaná verze) vznikla již v roce 2006 přesněji 19. 7. 2006. Tento program dokáže testy jak generovat a vyhodnocovat tak také exportovat do několika různých formátů. Na internetových stránkách autora je možnost program zakoupit nebo jen vyzkoušet v patnáctidenní zkušební verzi (Trial verze). Na této verzi je možno popsat základní vlastnosti a funkce. [5]

Při instalaci programu je nutné vypnout internetové prohlížeče, protože program si instaluje potřebné „pluginy“ pro spuštění testů přímo v prohlížeči. Po spuštění programu se objeví varování, že se jedná pouze o zkušební verzi programu a že testy pro komerční využití bude možné vytvářet až po zakoupení plné verze programu. Po potvrzení přejde program do jednoduchého menu, kde je zobrazeno kolik stojí plná verze programu podle toho, kde chce uživatel program užívat (Obrázek 2.5). Také si zde uživatel může prohlédnout tři tutoriály popisující základní manipulaci s programem. Zbytek možností slouží pouze pro otevření existujících testů nebo vytvoření nového. V položce „Help“ je možné spustit nápovědu (v angličtině), zmíněné tutoriály nebo se zde nalézá odkaz na internetovou stránku autora, kde jsou k prohlédnutí již hotové testy.



Obrázek 2.5: Úvodní obrazovka programu Quiz Builder

Při kliknutí na „Create new quiz“ se objeví editor, v němž se již nachází jedna prázdná otázka. Tu je možné vyplnit nebo smazat podle toho, jestli typ otázky odpovídá představě uživatele. Program podporuje osm druhů otázek: otázka typu „abcd“ s jednou nebo více správnými odpověďmi, otázka ano/ne, otázka s bankou slov, z nichž uživatel vybírá, přiřazování nebo seřazování slov, otázka na kterou musí být odpověď napsána nebo otázka, kde uživatel označí místo na obrázku.

Nad otázkou je lišta nástrojů pro základní ovládání testu pomocí níž je možné vkládat, mazat, přesouvat otázky, odpovědi a kategorie. Také je zde k dispozici slovník pro kontrolu gramatiky (bohužel jen v angličtině), nebo si tu uživatel může nastavit, aby program otázky mimo vypisování i vyslovoval (pouze ve verzi pro prohlížeč). U otázek a odpovědí má uživatel možnost změnit písmo nebo přiložit internetový odkaz a pro libovolné slovo otázky/odpovědi vyvolat jeho spuštění.

Naopak pod otázkami se nachází další sada možností pro kategorii či otázku. Pro kategorii otázek lze nastavit, jestli se bude automaticky měnit pořadí otázek, jestli budou z této kategorie do testu vybrány všechny otázky a také minimální procentuální zisk pro splnění kategorie. U otázky lze nastavit promíchání odpovědí, vyhodnocení otázky se záporným ohodnocením špatných odpovědí. Také se zde nachází volba pro vynucení vložení otázky do testu, přiložení obrázku či zvuku k jakékoli otázce nebo přiřazení časového limitu. Další možností je také přiložení takzvané „zpětné vazby“ tj. zobrazení obrázku nebo přehrání zvuku při odpovědi a to jednoduše při správné/špatné odpovědi nebo až při rekapitulaci zároveň se zobrazením správných odpovědí. S touto zpětnou vazbou je také zobrazena možnost nekonečného opakování otázky.

V hlavním menu je možno přejít na „Design“, kde se nastavuje vzhled testu pro prohlížeč. V tomto menu se nachází řada nastavení barev, velikostí a druhů písma, toho co bude program vypisovat, jaká budou zobrazena tlačítka a mnoho dalšího. V nabídce „Preview“ si je možné prohlédnout celý test a cvičně ho absolvovat. V „Properties“ se nastavují další možnosti jako je: zobrazení testovanému prospěl/neprospěl, přidělení výsledku do určité skupiny, povolení/zakázání patnáctiminutové pauzy, časový limit celého testu, datum vypršení testu, zda bude nutné se pro absolvování testu přihlásit, případně vyplnit i heslo, odesílání výsledků (na e-mail, webovému serveru nebo jen uložit v počítači) aj. Tyto výsledky se zobrazují v záložce „Reports“ a to s informacemi, které byli nastaveny. Pokud se ponechá původní nastavení, je každý absolvent testu pojmenován „Anonymous“. Exportování testu je velmi jednoduché, stačí stisknout

„Export options“, vybrat formát, do kterého se test převede (možnosti: exe, doc, swf), vybrat, kam se test uloží a poté stisknout „Export“.

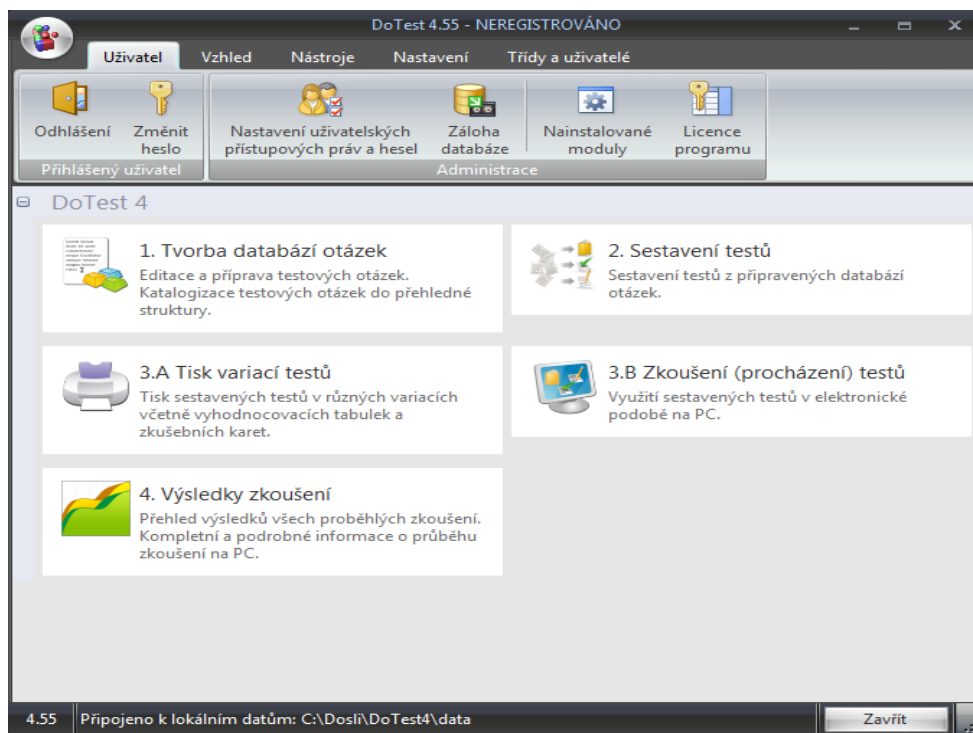
Hodnocení: program Quiz Builder 2.0.0 obsahuje mnoho funkcí pro generování testů, dokonce i možnost třídit otázky do skupin a z těchto skupin poté vybrat jen omezené množství otázek do testu. Export testu do formátu doc umožňuje i písemnou podobu testů (lze vytisknout vyplněnou i nevyplněnou verzi). Jako menší nevýhodu lze označit to, že vyplněná verze obsahuje všechny otázky, nikoliv jen ty, co byly vybrány do testu. Ale hlavním omezením je pak zpoplatnění programu (20. 10. 2010 byla cena verze pro školní účely, která by byla využívána jednou osobou 149 dolarů [5]).

2.6 DoTest 4.55

Další program, který je v rámci práce vyzkoušen, se nazývá DoTest 4.55. Jeho vydavatelem je tentokrát česká firma Dosli se sídlem v Opavě. Program slouží pro tvorbu databází otázek a z nich poté generuje testy, které je možno vytisknout nebo přímo absolvovat v rámci programu. Bohužel program je také komerčním SW, takže je možné vyzkoušet pouze tzv. „demonstrační režim“. Jedná se o třicetidenní zkušební verzi (opět Trial verze), ale během testování programu se žádná informace o tom, že by byl program nějakým způsobem časově omezen, neobjevila. Každopádně se na této verzi dají dobře popsat vlastnosti a funkce programu. [6]

Při instalaci programu si můžeme zvolit jak program nainstalovat. Jedná se o možnosti lokální instalace a síťové instalace (program podporuje vykonávání testů přes webový prohlížeč). Zajímavá je možnost instalovat program na USB disk a tím pádem mít možnost přenést si ho kam bude uživatel potřebovat.

Po spuštění programu je uživatel vyzván, aby vyplnil jméno a heslo. Toto jméno a heslo by měl obdržet e-mailem po zakoupení licence. Pro zkušební verzi jsou oba tyto údaje stejné, heslem i jménem je slovo „admin“. Poté se uživatel dostane do hlavního menu programu (Obrázek 2.6). Zde se nachází v horní liště možnosti nastavení a ve středu programu jsou již volby na přípravu testů.



Obrázek 2.6: Hlavní menu programu DoTest

Po stisknutí možnosti „Tvorba databází otázek“ se objeví jiné okno, v jehož levé části se zobrazují databáze otázek a pomocí ovládacího panelu nad nimi je možné databáze: vkládat (na stejnou či nižší úroveň), odebírat, přejmenovávat (v této položce lze i upravovat nastavení databáze), tisknout (jen danou databázi nebo i vnořené), vyhledávat, importovat nebo exportovat (volba exportovat nebyla povolena, zřejmě se jedná o omezení zkušební verze). Pokud uživatel chce vytvořit novou databázi otázek, klikne na příslušnou ikonu, zobrazí se okno, do kterého vyplní název, zda může jiný uživatel měnit tuto databázi, vyplní své jméno, požadovaný formát písma a také jestli budou body přiřazeny za otázku či za správnou odpověď (u některých typů otázek je možné mít více správných odpovědí).

Po vybrání některé databáze se v pravé části obrazovky zobrazí otázky, které k této databázi náleží. Otázky jsou rozděleny do pěti skupin: klasické (otázka a několik odpovědí), obrázkové (místo odpovědí obsahuje obrázky), přiřazovací (dva sloupce slov, které zkoušený musí spárovat), seřaďovací (termíny, které musí zkoušený správně uspořádat) a doplňovací (otázka, na kterou zkoušený odpoví písemně). Uživatel vybere skupinu a nad touto skupinou je opět ovládání, pomocí kterého lze provádět operace: vytvoření, smazání, upravení nebo kopírování otázky, také je tu volba „uložit změny“ a „storno“ pro potvrzení či zrušení úpravy otázky. Pokud uživatel chce vytvořit otázku,

klikne na volbu „nová“ a zobrazí se mu prázdná otázka, tu vyplní (nemusí se jednat pouze o jednoduchý text, je tu několik možností, jako změna formátu písma, vložení obrázku či jiného objektu), případně pokud je požadováno pokročilejší formátování, jako je vkládání tabulek či změna ohraničení, je tu možnost otevřít si pokročilejší editor.

Poté co uživatel vytvoří jednu nebo několik databází otázek, může přejít do menu „Sestavení testů“. Zde je opět v levé části seznam kategorií a v pravé seznam testů spadajících do této kategorie. Nad těmito možnostmi je opět ovládací panel na vkládání, odstraňování či změnu nastavení jak kategorií, tak testů. Při stisknutí na tlačítko „Nový test“ se zobrazí menu, ve kterém si nastavíme požadované vlastnosti: jméno testu, publikace (zda se zobrazí i ostatním uživatelům programu), omezení (maximální počet spuštění, časové omezení, spuštění pouze v aplikaci či pouze přes www prohlížeč a jiné), poznámka k testu, potvrzení zda se po vykonání testu zobrazí ihned výsledky, seřazení otázek podle obtížnosti, či mixování vzhledem k databázím a další možnosti. Poté je nutné do testu vybrat určité databáze otázek a těm nastavit, zda určitá otázka v této kategorii musí být vybrána či musí být ignorována.

Když již existuje nějaký test, tak existují dvě možnosti. Zvolit „Tisk variací testů“, zde se zobrazí menu, v jehož spodní části se vybere test a v horní části je nutné stisknout variantu „Generovat a tisknout nové variace“. Zde se podle vytvořené šablony vytvoří již konkrétní test nebo více (záleží na naší volbě). Zobrazí se jeho/jejich náhled a to ve čtyřech různých formátech (jeden, dva, tři sloupce nebo učebnice). Uživatel si vybere jednu variantu a tu pak může tisknout nebo exportovat do PDF. Poté se může vrátit do menu s vytvořenými variantami a vytisknout si vyhodnocovací tabulky (to jsou tabulky, ve kterých je vypsáno, jak má test být správně vyplněn) a zkoušecí karty (to jsou malé karty, na které zkoušení vyplní svá jména, třídu, variantu testu, datum a jejich odpovědi). Druhou možností je „Zkoušení (procházení) testů“, zde program náhodně vygeneruje jednu variantu podle šablony a tu dá zkoušenému přímo k dispozici. Pro malé děti je tu varianta, kdy správnou odpověď vybírají hrou. V položce „Výsledky zkoušení“ poté může osoba s určitými právy vidět, kdo, kdy, jaký test a s jakým výsledkem vykonal, případně si tato data vytisknout.

Jak již bylo zmíněno, v horním panelu se nachází nastavení programu. Zde se může každý uživatel odhlásit, změnit si heslo, případně si změnit vzhled okna (vybírá si z předem definovaných možností). Další možnosti už závisí na stupni oprávnění. Jsou tu tři základní skupiny: Student, Teacher a Administrator. Administrátor programu může

jednotlivým skupinám pravomoci přidávat a také ubírat, zálohovat databáze, spravovat přidané moduly, měnit licenci programu (podle zakoupené), vkládat obrázky do knihovny, nastavovat www rozhraní (pokud se jedná o síťovou verzi), nastavovat princip přihlášení (jestli má každý student své jméno a heslo nebo všichni používají jedno heslo) a pak samozřejmě vkládá studenty, zaměstnance a ty ještě může rozdělovat do tříd (třída má svého třídního učitele a studenty).

Shrnutí na závěr: výhodou je členění otázek do kategorií a poté možnost do testu náhodně vybrat určitý počet otázek z této kategorie. To umožňuje vytvářet stále různé testy, podle jedné šablony. Možnost tisknout test prázdný i test vyhodnocený. Program je sice zaměřen na písemné zkoušení pomocí „zkoušecích karet“. Není to ovšem nutné, na testu je dost prostoru pro vyplnění jména a data, takže lze každému studentovi dát svůj vlastní test. Studenty je možné zkoušet přímo na PC. Manipulace s programem je velmi jednoduchá a přehledná, jak se vše ovládá a funguje, zjistí každý na první pohled bez složitostí. Nevýhodou je absence porovnávání vygenerovaných variant, ale tuto vlastnost neměl žádný z testovaných programů. A dalším mínusem je zpoplatnění programu (verze pro jednu osobu se školním zvýhodněním, bez webového rozhraní, stála dne 22. 10. 10 přesně 1210 Kč s DPH [6]).

3 Cíl práce: požadovaná funkčnost programu

Navrhnout a naprogramovat vlastní aplikaci pro generování zkouškových testů je velmi obecné zadání problému. Z pohledu na popsání existujících programů je patrné, že aplikace může obsahovat mnoho nejrůznějších nástrojů, které usnadňují práci tvůrce testu. Jestliže se problém omezí pouze na vytváření, tisk a archivaci testů, je stále možné některé části programu vytvořit několika způsoby a tím například omezovat nebo ztěžovat práci uživatele. Proto je třeba stanovit základní funkcionalitu, kterou by měl program splňovat. Po konzultaci s vedoucím práce byla stanovena tato interakce uživatele a programu:

- 1) Aby bylo možné vygenerovat test, je nutné do programu vložit databázi otázek. Tento problém lze řešit dvěma způsoby. Databáze otázek může být realizována skutečným databázovým systémem, kde je nutné vytvořit editor pro vkládání otázek do databáze, nebo lze k tomuto účelu využít textový soubor, jehož editace je možná kdykoliv a není třeba žádný další SW. Možná struktura souboru vypadá následovně:

```
*Jaký je správný zápis čísla 58 v binární číselné soustavě?  
-101010b  
+111010b  
-100100b  
-110011b
```

Hvězdička označuje otázku, plus správnou odpověď a mínus špatnou odpověď. V každém souboru je skupina otázek týkající se jednoho uceleného tématu. Počet otázek v souboru není omezen a počet souborů je omezen pouze počtem otázek v testu.

- 2) Uživatel zvolí počet otázek v testu (např. deset, dvanáct nebo patnáct), každé otázce přiřadí databázi otázek a nastaví její parametry. Parametrem otázky je požadovaný počet odpovědí (např. čtyři) a také kolik z těchto odpovědí má být správných (žádná, jedna, maximálně dvě atp.).
- 3) Dále přichází na řadu generátor testu, který náhodně vybere z příslušného souboru otázku a požadované odpovědi (např. tři špatné a jednu správnou). Poté náhodně změní pořadí odpovědí a po vygenerování všech otázek náhodně změní i jejich pořadí (aby např. otázka k okruhu číslo jedna byla v testu pokaždé pod jiným číslem).

- 4) Tento vygenerovaný test program uloží do databáze testů, aby s ním bylo možné později pracovat. Databázi testů lze implementovat pomocí textového souboru či vybraného databázového systému. V tomto případě není vzhledem k pozdějšímu zpracování testu textový soubor ideálním řešením, proto je nutné zvolit jeden z dostupných databázových systémů. Ten by vzhledem k povaze aplikace měl být spustitelný na lokálním stroji a neměl by být zpoplatněný.
- 5) Dalším krokem je využití generátoru tiskových sestav. Uživatel si vytvoří individuální tiskovou šablonu (např. rozložení testu na výšku / na šířku, vložení hlavičky, případně vložení zápatí s nápovědou apod.). Tu bude možné uložit pro pozdější využití (např. pro stejný předmět, kde bude nutné změnit pouze datum zkoušky). Po vytvoření (načtení) této šablony, uživatel vybere test, který chce publikovat a zvolí jeho formát (např. počet sloupců s otázkami). Program podle tohoto nastavení vytiskne jak požadované množství nevyplněných testů, tak také jeden správně vyplněný pro kontrolu.
- 6) Vše co bylo popsáno v předešlém textu, tvoří základní funkcionalitu programu. Vhodným rozšířením je vytvoření funkce pro porovnání aktuálního testu s již vytvořenými testy v databázi. Tuto podobnost vyjádřit v procentech a vypsát ty nejpodobnější.

Výběr programovacího jazyka ani vývojového prostředí není nikterak omezen. Nejlepší volbou jsou programovací jazyky, které jsou vyučovány přímo na Technické univerzitě v Liberci, mezi které patří C++, C#, Java, Pascal aj. Z vývojových prostředí lze zvolit například Delphi, Visual Studio Express nebo některou „open source“ alternativu (DevC++, FreePascal, Lazarus ...). Program musí fungovat na Windows XP, Windows Vista a Windows 7. Výhodou by byla přenositelnost i na ostatní operační systémy, ale není to podmínkou.

4 Nástroje použité při tvorbě vlastního řešení

4.1 Vývojové prostředí wxDev-C++ 7.3.1

Integrované vývojové prostředí (IDE) wxDev-C++ je určeno pro tvorbu aplikací v jazyce C/C++. Jedná se o rozšíření projektu Dev-C++, který byl vyvíjen Colinem Laplacedem a jeho spolupracovníky do roku 2005. [7] Prostředí wxDev-C++ je rozšířeno o návrhové nástroje pro tvorbu aplikací s hlavními a dialogovými okny, která vytváří prostřednictvím grafické knihovny wxWidgets.

Designér oken v prostředí wxDev-C++ umožňuje sestavovat okna jednoduchým „přetahováním“ grafických prvků (tlačítka, textová pole atd.) do vzorového okna. V tomto designéru je také možné měnit jejich vlastnosti a přiřazovat jim události. Tyto události jsou designérem propojeny s funkcemi, které jsou vytvořeny pomocí editoru jazyka C++. Takto vytvořená okna jsou včetně všech grafických prvků přeložena do jazyka XRC. XRC je značkovací jazyk, který je založený na XML a využívaný grafickou knihovnou wxWidgets pro zaznamenávání oken. Takto uložené okno může být načteno při spuštění programu nebo být přeloženo do libovolného programovacího jazyka. Tento postup také umožňuje přenositelnost oken na různé operační systémy. [7]

Editor jazyka C/C++ je vybaven obvyklými nástroji jako je průzkumník tříd, „našeptávač“, různé nastavování projektu (např. připojování knihoven, automatické nastavování verzí aj.), zvýraznění syntaxe jazyka atd. Je zde také rozšířená podpora ladění, která umožňuje sledovat hodnoty určité proměnné za běhu programu, výpis lokálních proměnných, automatické sledování stavu zásobníku, zobrazování registrů procesoru a také podporu pro GDB. GNU project debugger (GDB) umožňuje program zastavovat ve specifických případech, spouštění programu s různými parametry apod. GDB je hojně využíván především na UNIXových systémech, ale i v systémech Windows. Prostředí WxDev-C++ využívá kompilátor MingW, podporuje i Microsoft VC++ a některé další kompilátory. [7]

Tým, který prostředí wxDev-C++ vytváří: Guru Kathiresan (tvůrce projektu), Tony Reina, Esteban Aguilar (administrátoři projektu), Malcolm Nealon, Kin Ming, Nuklear Zelph (hlavní vývojáři) a Edward Toovey (tvůrce dokumentace). Záměrem autorů je poskytnout vývojářům využívajícím knihovnu wxWidgets nástroj s otevřeným zdrojovým kódem, který je volně ke stažení. [7]

4.2 Databázový systém MySQL 5.5

Databáze je strukturovaná kolekce dat (např. seznam výrobků v obchodě, galerie obrázků, údaje uchovávané v podnikové síti apod.). Pro přidávání a zpracování těchto dat, je nutné používat databázový systém, jako je například MySQL (relační databázový systém). Pojem relační databáze označuje databázi, která je strukturována do různých tabulek, což je výhodnější, než aby byla všechna data v jednom velikém úložišti (zejména kvůli rychlosti a flexibilitě). [8]

MySQL je systém vyvíjený a distribuovaný firmou Oracle Corporation, která k němu také poskytuje uživatelskou podporu. MySQL je open source SW, tudíž je možné získat ho bezplatně z internetu a ihned používat. Uživatel může studovat jeho zdrojové kódy a poté je případně i modifikovat tak, aby vyhovovaly jeho požadavkům. MySQL pro tyto účely využívá licenci GPL (GNU General Public License). Pokud ovšem uživateli licence GPL nevyhovuje nebo chce MySQL využívat pro komerční účely, může si od firmy Oracle zakoupit oficiální licenci pro komerční využití SW. [8]

Název systému je odvozen od jazyka, který je používán pro komunikaci s tímto databázovým systémem, tedy SQL (Structured Query Language). SQL lze do češtiny přeložit jako strukturovaný dotazovací jazyk. SQL je jazyk standardizovaný normami ANSI/ISO SQL. Tento standard je vyvíjen již od roku 1986 a existuje několik jeho verzí (1992, 1999, 2003). Název je doplněn slovem „My“, které představuje jméno dcery zakladatele tohoto projektu (Monty Widenius). [8]

Systém MySQL je napsán v jazyce C a později C++. Může pracovat na velkém množství systémů (Windows, Linux, FreeBSD, Solaris, Mac OS atd.). Pro zajištění této portability jsou využity nástroje CMake (dřívější verze využívaly např. Autoconf). Systém MySQL je strukturován tak, aby ho bylo možné použít v síťových aplikacích klient/server, stejně tak dobře ho lze ovšem použít jako knihovnu pro aplikace, které nemají přístup k síti. [8]

Systém podporuje velké množství datových typů: signed/unsigned integer, float, double, char, varchar, text, date, time, datetime, timestamp, set, enum atd. Pro práci se systémem lze využít standardních funkcí SQL (select, insert, delete, update ...). Systém podporuje také agregační funkce (min, count ...), obsahuje celou řadu možností pro spojování tabulek a mnoho dalších nástrojů a funkcí. [8]

Aplikace, které budou MySQL využívat, mohou být napsány v mnoha různých programovacích jazycích (C, C++, Java, Perl, PHP, Python, Ruby...). Pro většinu z nich má MySQL dostupné své API a pro ty ostatní je možné doplnit systém o požadovaný connector. Systém podporuje celou řadu znakových sad, takže u něho odpadá problém s národními znaky. Chybová hlášení jsou k dispozici ve velkém množství jazyků, záleží na nastavení uživatele. [8] Všechny vlastnosti a funkce systému jsou zdokumentovány v angličtině a některé také v dalších pěti jazycích. Tato dokumentace je dostupná na internetové adrese <http://dev.mysql.com/doc/>.

4.2.1 Použití MySQL v prostředí wxDev-C++

Pokud chce programátor používat databázový systém MySQL v aplikaci, která je napsána v jazyce C++ je nutné tyto dvě aplikace propojit. Propojení lze provést dvěma způsoby. Jedním ze způsobů je Connector C++ a druhým MySQL C API.

Použití Connectoru C++ je „čistější“ varianta, protože zde není potřebné volat žádné funkce napsané jazykem C. Tento Connector podporuje standardní průmyslové API s názvem JDBC 4.0 a také schéma objektově orientovaného programování. Měl by usnadnit a také zrychlit vývoj aplikace. Opět je k dispozici pod dvěma druhy licencí: GPL (s výjimkou FLOSS) a komerční licence. Nicméně manuál k tomuto Connectoru je zaměřen na vývoj ve Visual Studiu (Windows) a NetBeans (Linux). [8]

Druhá varianta je standardní MySQL C API. To je distribuováno již v samotné instalaci MySQL serveru. Obsahuje dvě knihovny, statickou knihovnu „mysqlclient.lib“ nebo dynamickou knihovnu „libmysql.dll“, přes které se lze s MySQL serverem spojit. Jestliže se programátor rozhodne použít statickou knihovnu, může dojít k chybě, pokud je vytvářená aplikace kompilována jiným kompilátorem než tato statická knihovna. [8]

V prostředí wxDev-C++ může dojít k chybě i při použití dynamické knihovny. Proto je nutné vytvořit si vlastní statickou knihovnu „libmysql.a“ pomocí programů pexports a dlltool. [8]

4.3 Knihovna wxWidgets

WxWidgets je knihovna jazyka C++ umožňující vytvářet grafické aplikace na systémech Windows, OS X, Linux a Unix (na 32bitových i 64bitových verzích). Stejně tak je možné vytvářet jejím prostřednictvím aplikace pro mobilní systémy (Windows Mobile, iPhone SDK atd.). Na jejím základě jsou vytvořeny grafické knihovny i pro jiné jazyky, např. Python (název knihovny wxPython), Ruby (wxRuby), Perl (wxPerl) a řadu

dalších jazyků. Na rozdíl od jiných přenositelných nástrojů využívá systémové API, takže její pomocí lze vytvářet aplikace s přirozeným vzhledem (jen málo nástrojů je závislých na platformě, protože chybějící funkce jsou často emulovány). [9]

Knihovna wxWidgets, která je volně ke stažení, je open source SW. Současná licence knihovny je „wxWindows Licence“. Nyní se čeká na schválení „wxWidgets Licence“, která by měla být totožná pouze s odlišným názvem. WxWindows Licence je v podstatě licence „Lesser General Public License“ (L-GPL) s výjimkou, že odvozené práce mohou být šířeny za podmínek, které si stanoví sám tvůrce. To znamená, že sám tvůrce si stanoví, jestli bude pomocí knihovny wxWidgets vytvářet SW pod licencí GPL nebo SW komerční. [9]

Knihovna byla vytvořena Julianem Smartem v roce 1992 a cílem bylo vytvořit nástroj, který by fungoval jak na stanicích se systémem Unix, tak na stanicích se systémem Windows. Komunita podílející se na vývoji této knihovny se postupem času zvětšovala (osoby, které se na tvorbě knihovny významně podílely, jsou uvedeny na internetových stránkách projektu). V současné době je knihovna wxWidgets přijata jako alternativa k zpoplatněným nástrojům s nezveřejněným zdrojovým kódem od společnosti Microsoft. Knihovna je považována za nejrychlejší a nejsnazší možnost, jak psát aplikace přenositelné na operační systémy Unix, Linux nebo Mac. [9]

Knihovna wxWidgets poskytuje velké množství obvyklých grafických prvků, jako jsou tlačítka, textová pole, radio-buttony, check-boxy a podobně, ale také prvky pokročilé, jako jsou wxTreeCtrl (hierarchicky strukturovaná textová pole), wxToolbar (lišta s nástroji), wxGrid (tabulka) a další nástroje. Všechny tyto prvky jsou doplněny událostmi, které lze staticky (při kompilaci) nebo dynamicky (za běhu programu) propojit s členskými funkcemi. Dále je tu celá řada nástrojů, které lze využít k tisku dokumentů v systému Windows nebo generování PostScriptu na Unixových systémech. Tyto i další nástroje jsou detailně popsány v manuálech čítajících okolo 1800 stran. Zmíněné manuály je možné zobrazit v několika formátech (HTML, PDF a další). [9]

5 Program pro Generování Testů (vlastní řešení)

V několika následujících kapitolách je popsána implementace modulů, které zajišťují výše zmíněnou funkcionalitu (kapitola 3) a tím umožňují uživateli vytvořit tiskovou sestavu v požadovaném formátu. V kapitole 5.1 jsou zmíněny některé nástroje nebo termíny, které jsou v práci používány.

5.1 Použité nástroje

5.1.1 Nástroje wxDC

Nástroje wxDC slouží k vykreslení různých tvarů (některé tvary viz Tabulka 2) na vybraná místa v okně i mimo okno v závislosti na tom, který nástroj wxDC uživatel zvolí (příklady viz Tabulka 1). [10]

Tabulka 1: Vybrané nástroje wxDC [10]

DC nástroj	K čemu lze tento nástroj použít
wxClientDC	Kreslení tvarů po uživatelské části okna
wxWindowDC	Kreslení tvarů po celém okně, včetně okrasných oblastí
wxScreenDC	Kreslení tvarů na celou obrazovku / kopírování tvarů z celé obrazovky
wxMemoryDC	Kreslení do bitmapy / kopírování z bitmapy
wxPrinterDC	Kreslení tvarů pro tiskárnu
wxPostScriptDC	Kreslení tvarů pro soubor nebo tiskárnu PostScript

Před kreslením většiny tvarů je možné nastavit si určité vlastnosti, jako jsou například: styl písma (procedura `SetFont`), barva písma (`SetTextForeground`), barva pozadí písma (`SetTextBackground`), barva výplně prvku (`SetBrush`), aktuální barva pozadí (`SetBackground`) a řada dalších. [10]

Tabulka 2: Vybrané metody nástroje wxDC [10]

Metoda třídy wxDC	Objekt, který je vykreslen
<code>Clear</code>	Celý objekt wxDC je překreslen aktuální barvou pozadí
<code>DrawCircle</code>	Vykreslení kružnice
<code>DrawText</code>	Vykreslení požadovaného textu
<code>DrawRectangle</code>	Vykreslení čtverce
<code>DrawCheckMark</code>	Vykreslení symbolu „fajfky“ (symbol správně odpovědi)
<code>DrawLine</code>	Vykreslení čáry
<code>DrawEllipse</code>	Vykreslení elipsy

5.1.2 Eventy (události)

Event je událost, která vzniká nějakou změnou v systému, nejčastěji se jedná o určitou uživatelskou akci v dialogovém nebo hlavním okně (např. stisk tlačítka). Této události lze přiřadit rutinu, která bude vyvolána jako reakce (tzv. událostmi řízené programování). Spojení události a příslušné rutiny se provádí pomocí tabulky eventů, jež se musí nacházet v každé třídě, která bude eventy využívat (např. designér prostředí wxDev-C++ automaticky vytváří event vyvolaný po zavření okna).

Eventy se dělí podle druhů (wxMouseEvent, wxEraseEvent atd.) a každý druh odpovídá určitému původu události (např. wxMouseEvent jsou eventy vyvolané pomocí myši uživatele – kliknutí, pohyb). Některé druhy eventů je nutné upřesnit výběrem typu (např. wxCommandEvent). Takové eventy je poté možné využít k vytvoření vlastního eventu, upozorňujícího na nějakou požadovanou událost. Vlastní typ eventu je možné definovat těmito příkazy: [11]

- V deklarační oblasti příkazem:

```
DECLARE_EVENT_TYPE(typ_eventu, id_eventu)
```

- V definiční oblasti příkazem: DEFINE_EVENT_TYPE(typ_eventu)

V těle funkce poté může být vytvořen event požadovaného typu, mohou mu být dány určité parametry a poté ho lze předat libovolnému oknu/třídě. Toto se provádí následujícími příkazy: (druh eventu je wxCommandEvent) [11]

```
wxCommandEvent event(typ_eventu, id_eventu);  
  
event.SetInt(int);  
  
event.SetString(wxString);  
  
event.SetExtraLong(long);  
  
název_třídy->GetEventHandler()->AddPendingEvent(event);
```

Ve třídě, která má na tuto událost reagovat, musí být vytvořena tabulka eventů s jednoznačným přiřazením reakce. Vytvoření tabulky eventů: [11]

- V deklarační oblasti třídy (privátní vlastnost) příkazem:

```
DECLARE_EVENT_TABLE()
```

- V definiční oblasti příkazem:

```
BEGIN_EVENT_TABLE(třída, rodič_třídy)
    EVT_LEFT_DOWN(název_reakce) //levé tlačítko
    EVT_RIGHT_DOWN(název_reakce) //pravé tlačítko
    EVT_ERASE_BACKGROUND(název_reakce) //překreslení pozadí
    EVT_COMMAND(id_eventu, typ_eventu, název_reakce)
        // libovolný vytvořený event
END_EVENT_TABLE()
```

5.1.3 Zmíněné datové typy

Při popisu tříd, respektive jejich metod a atributů je v některých případech uveden datový typ. U těch nejčastěji používaných je proto vhodné vysvětlit, k čemu zmíněný datový typ slouží. U každého datového typu je napsáno k čemu slouží a do závorky je umístěna informace o jeho původu:

- bool – uchování hodnot true/false (standardní datový typ jazyka C++)
- int – uchování celého čísla (standardní datový typ jazyka C++)
- long – uchování celého čísla (větší než int, datový typ jazyka C++)
- uint = unsigned int – uchování celého čísla bez záporných hodnot (standardní datový typ jazyka C++)
- float – uchování desetinného čísla (standardní datový typ jazyka C++)
- wxString – uchování řetězce (knihovna wxWidgets)
- wxArrayInt – dynamické pole celých čísel (knihovna wxWidgets)
- wxStringArray – dynamické pole řetězců (knihovna wxWidgets)
- wxFont – uchování všech informací o písmu (knihovna wxWidgets)
- wxColour – uchování informace o barvě (knihovna wxWidgets)
- wxPoint – uchování souřadnice – x, y souřadnice (knihovna wxWidgets)
- wxChoice – ComboBox – grafický prvek pro výběr hodnoty ze seznamu řetězců, seznam se zobrazí po kliknutí na prvek (knihovna wxWidgets)
- wxSpinCtrl – grafický prvek ve kterém lze pomocí šipek vybrat hodnotu z určitého číselného intervalu (knihovna wxWidgets)
- wxTextCtrl – grafický prvek pro zápis řetězce (knihovna wxWidgets)
- wxStaticText – grafický prvek pro zobrazení popisku (knih. wxWidgets)

5.1.4 Další pojmy

Dialogové vs. hlavní okno

Rozdíl mezi dialogovým a hlavním oknem je ten, že dialogová okna zablokuje interakci s původním programem do té doby, než na něho uživatel zareaguje (tzv. modální okno). Toho se využívá zejména pro předání informace uživateli nebo k výzvě, aby reagoval na nějakou vzniklou událost.

Vytvoření prototypu dynamického pole

Pro libovolnou strukturu či třídu je v prostředí wxDev-C++ možné vytvořit dynamické pole těchto struktur. K tomu je nutné vytvořit prototyp („nový datový typ“). Definice prototypu dynamického pole se provádí ve dvou krocích: [11]

- V deklarační oblasti příkazem:

```
WX_DECLARE_OBJARRAY(název_třídy/struktury, název_pole);
```

- V definiční oblasti příkazem: (nutný je import knihovny wx/arrimpl.cpp)

```
WX_DEFINE_OBJARRAY(název_pole);
```

Pojem metoda

Při popisu tříd je využit termín „metoda“. Ten by se v souvislosti s jazykem C++ neměl používat, nicméně v tomto textu zobecňuje následující pojmy:

- „konstruktor“ – rutina volaná automaticky při vytvoření instance třídy
- „destruktor“ – rutina volaná automaticky při likvidaci instance třídy
- „procedura“ – rutina, která nevrací žádný výsledek, pouze mění atributy objektu/třídy
- „funkce“ – rutina, která na svém konci vrací konkrétní hodnotu
- „funkce GET“ – rutina, jejímž výstupem je hodnota určeného atributu
- „procedura SET“ – rutina, která nastaví hodnotu určeného atributu

5.2 GUI pro nastavení parametrů testu

5.2.1 Třída „souborOtazek“

Tato třída zajišťuje zaznamenání souboru, který obsahuje databázi otázek. Třída se skládá ze tří atributů a šesti metod. Atributy třídy jsou: cesta k zadanému souboru v souborovém systému (datový typ wxString), název kapitoly zaznamenané v souboru (wxString) a informace o existenci či neexistenci souboru (bool).

V **konstruktoru** třídy je provedena kontrola existence tohoto souboru pomocí třídy `wxFile` (statická funkce `Exists`). Pokud soubor existuje, je z něho pomocí tříd `wxFileInputStream` a `wxTextInputStream` přečten první řádek a ten je uložen jako název tématu, ke kterému náleží otázky v souboru. Funkce s názvem ***PopisekSouboru*** ověří, zda soubor stále existuje a pokud ano je z něho přečten první řádek, který se ignoruje. Poté se čtou další řádky do té doby, než se narazí na první otázku. Výsledkem této funkce je popisec, který řekne, že soubor již neexistuje, soubor se nepovedlo otevřít, soubor neobsahuje otázky (v případě, že by se při čtení souboru došlo na jeho konec), soubor nemá uveden popisec nebo popisec zapsaný v tomto souboru.

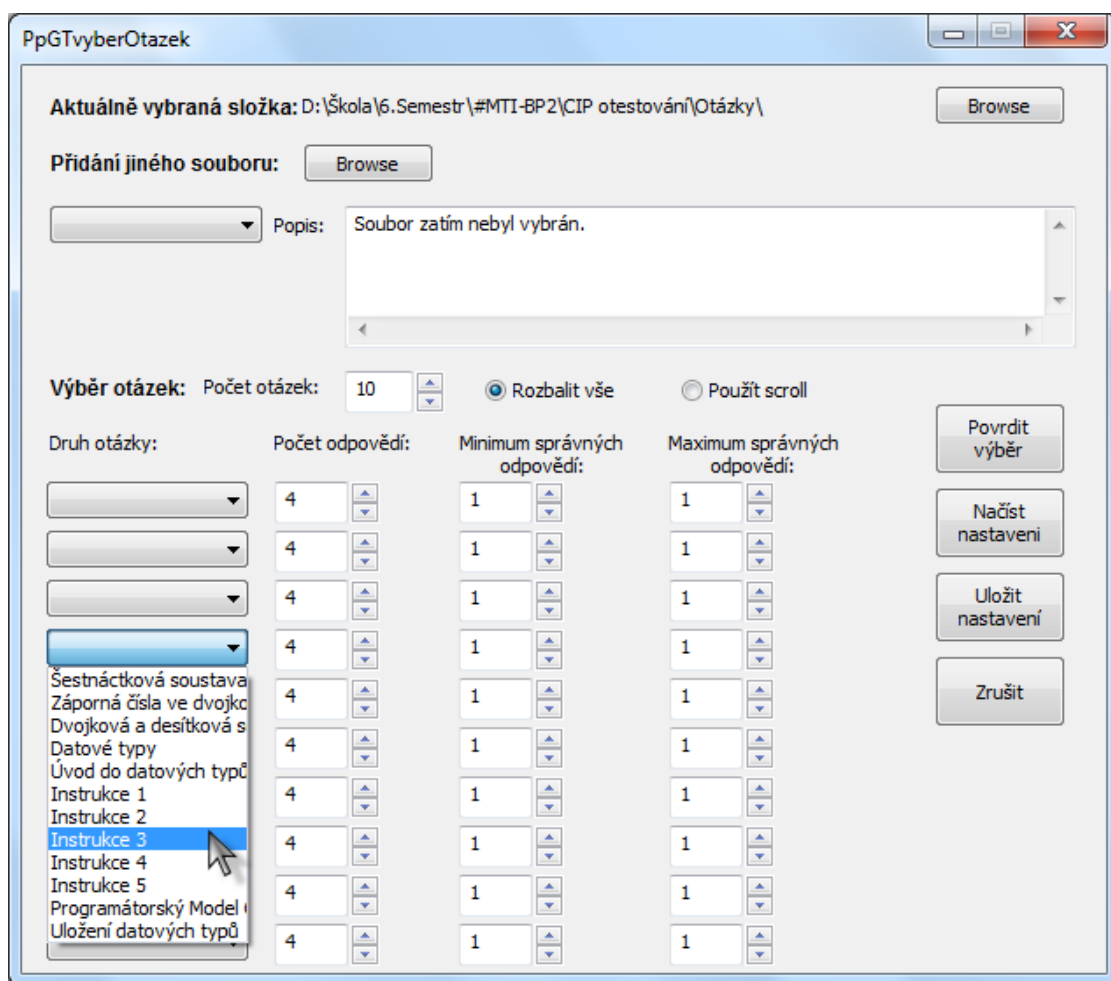
5.2.2 Třída „modulOtazky“

Pomocí této třídy je realizováno zobrazování proměnlivého počtu grafických prvků, kterými se nastavují vlastnosti otázky v okně „dlgVyberOtazek“ (5.2.3). Třída obsahuje čtyři atributy a deset metod. Atributy jsou grafické prvky pro výběr okruhu otázky (`wxChoice`), pro volbu počtu odpovědí a pro volbu minimálního a maximálního počtu správných odpovědí (`wxSpinCtrl`).

Konstruktor této třídy zobrazí zmíněné grafické prvky a k tomu je nutné zadat tři parametry – okno a dvě souřadnice. Hodnoty, které je možné vložit do prvků `wxSpinCtrl` jsou omezeny (2–6 odpovědí, 0–6 správných odpovědí). Funkce ***GetStav*** vrací řetězec (`wxString`) složený z aktuálních hodnot vybraných v grafických prvcích (čtyři hodnoty typu `int` odděleny středníkem). Procedura ***SetStav*** má naopak na vstupu řetězec, který pomocí třídy `wxStringTokenizer` rozdělí na čtyři části (dělení řetězce proběhne v místě, kde je středník). Tyto hodnoty poté převede na datový typ `int` a upraví jimi aktuální hodnoty ve všech prvcích.

5.2.3 Dialogové okno „dlgVyberOtazek“

V tomto dialogovém okně (Obrázek 5.1) se provádí obsahové nastavení testu. Určí se počet otázek v testu a každé otázce lze přiřadit soubor s databází otázek a také počet odpovědí (včetně určení intervalu správných odpovědí). Okno je vytvořeno pomocí designéru prostředí `wxDev-C++`. Funkcionalita je zajištěna reakcemi na eventy, doplněnými o šest atributů a dvě metody. Doplnujícími atributy jsou: pole souborů s otázkami (souborOtazek 5.2.1), indikátor velikosti tohoto pole (`int`), počet skutečně načtených souborů (`int`), pole modulů s vlastnostmi (modulOtazky), indikátor počtu modulů (`int`) a výstup tohoto okna (pole objektů, které jsou popsány v kapitole 5.4.1).



Obrázek 5.1: Grafická podoba okna „dlgVyberOtazek“

Konstruktor třídy provede sestavení okna a inicializaci doplňujících atributů (maximální počet otázek v testu je omezen na 15, maximální počet načtených souborů není omezen). **Destruktor** třídy likviduje pouze doplňující atributy. Grafické prvky okna jsou zlikvidovány již při uzavření okna tak, aby se samotnou třídou bylo možné dále pracovat (např. získání výstupní hodnoty).

Při stisknutí tlačítka, které mění aktuálně vybranou složku, se zobrazí dialogové okno pro výběr složky, toto je realizováno grafickým prvkem `wxDirPickerCtrl`. Pokud dojde k volbě nové aktuální složky, jsou vymazány všechny aktuálně načtené soubory (z pole, které je uchovává), do popisku je zobrazena nově vybraná složka (cesta v souborovém systému) a poté je prostřednictvím datového typu `wxDir` pro všechny soubory s koncovkou „.txt“ vyvolána procedura **PridejPrvekDoPole**. Tato procedura vytvoří objekt `souborOtazky` pro zadaný soubor a pokud se soubor povedlo úspěšně otevřít, přidá jej do pole souborů s otázkami a do všech prvků `wxChoice` v okně uloží obsah prvního řádku (název tématu v souboru). Tlačítko pro výběr jednotlivého souboru

je realizováno grafickým prvkem `wxFilePickerCtrl`. Po stisku je také otevřeno dialogové okno a po vybrání souboru spuštěna procedura `PridejPrvekDoPole`.

Pokud je v grafickém prvku `wxChoice` vybráno téma, je zavolána funkce pro získání popisku ze souboru (funkce třídy `SouborOtazek`) a ten je zobrazen do grafického prvku `wxTextCtrl`. Moduly pro popis vlastností otázek jsou vloženy v grafickém prvku `wxScrolledWindow`. Tomuto prvku lze nastavit dvojce rozměry – fyzické a virtuální. Pokud je fyzický rozměr menší než virtuální, může dojít k zobrazení „posouvátka“. Toto umožňuje dvojí zobrazení okna tak, že zobrazuje všechny moduly nebo zobrazí pouze omezenou část modulů, ale pomocí zmíněného „posouvátka“ se lze dostat i k těm ostatním. Mezi těmito možnostmi lze přepínat pomocí prvků `wxRadioButton` s názvy „rozbalit vše“ / „použít scroll“. Pokud dojde ke změně hodnoty prvku `wxSpinCtrl`, který zobrazuje počet otázek testu, jsou vytvořeny/zlikvidovány moduly otázek (objekty třídy `ModulOtazky`) v poli k tomu určeném a musí být nastavena správná virtuální velikost grafického prvku `wxScrolledWindow` a pokud je zvoleno zobrazování „rozbalit vše“ nastaví se i správná fyzická velikost.

Posledními grafickými prvky okna jsou čtyři tlačítka. Při stisku tlačítka „Zrušit“ je okno uzavřeno. Při stisknutí tlačítka „Uložit nastavení“ je zobrazeno okno pro výběr souboru (grafický prvek `wxFileDialog`). Pokud uživatel vybere soubor, je mu přidána koncovka „.sav“ (pokud ji již neobsahuje) a pomocí nástrojů `wxFileOutputStream` a `wxDataOutputStream` je do něho uložena následující posloupnost: řetězec pro identifikaci toho, že se jedná o soubor tohoto programu („PpGTulozeneNastaveni“), cesta k aktuální složce, počet načtených souborů, cesty ke všem načteným souborům, aktuální počet otázek v testu a hodnoty v modulech s otázkami (výstup funkce `GetStav` třídy `ModulOtazky`). Po stisku tlačítka „Načíst nastavení“ uživatel také vybírá soubor (`wxFileDialog`). Pokud uživatel skutečně soubor vybere, je ověřena jeho koncovka („.sav“) a pomocí nástrojů tříd `wxFileInputStream` a `wxDataInputStream` je načten první řetězec. Pokud se shoduje s řetězcem „PpGTulozeneNastaveni“ jsou vymazány všechny aktuální soubory a obsahy prvků `wxChoice`. Dále se přečte řetězec s názvem složky (pouze pro zobrazení). Poté se načtou cesty k souborům (nejdříve počet, poté daný počet řetězců) a pokud všechny soubory existují, jsou načteny i stavy modulů (existují-li jen některé soubory, jsou ponechány výchozí stavy). Tlačítko „Potvrdit výběr“ zajišťuje vytvoření výstupu okna, tedy vytvoření pole objektů třídy „popisProGenOtazky“ (kapitola 5.4.1). Pro každý modul otázky jsou vyhodnoceny jeho údaje: zda je vybrán

soubor s otázkami, je-li počet odpovědí větší/roven maximum správných odpovědí a také jestli je maximum správných odpovědí větší/rovno minimu. Pokud jsou tyto požadavky splněny pro všechny moduly (jsou vytvořeny všechny objekty) je okno uzavřeno. Nesplňuje-li některý modul tyto požadavky, je o tom uživatel informován a je ukončeno další procházení (předtím vytvořené objekty jsou vymazány).

5.3 Práce s databází testů

5.3.1 Třída „praceMySQL“

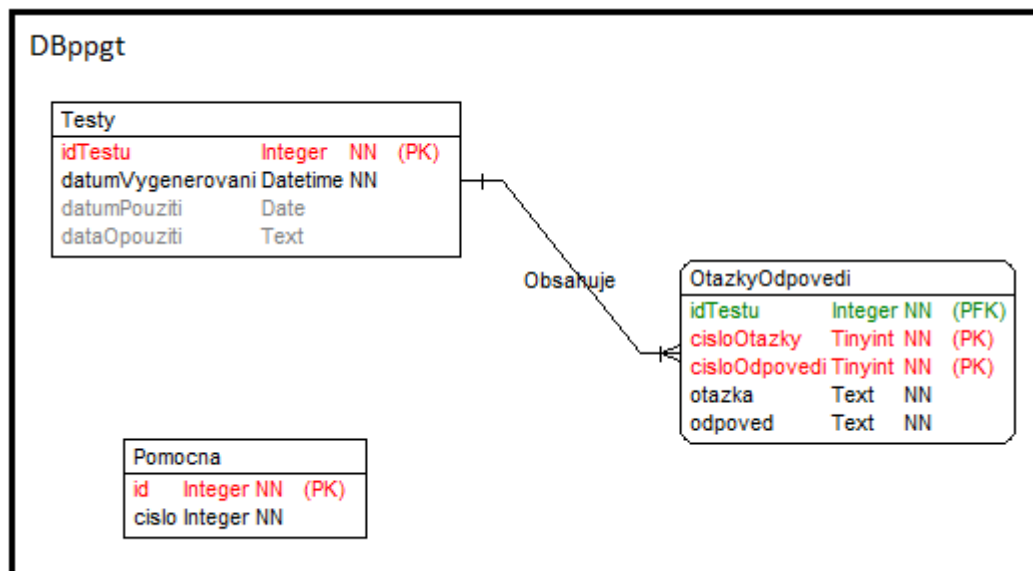
Tato třída realizuje spojení a všechny potřebné úkony v databázovém systému. Třída obsahuje tři atributy a dvanáct metod. Atributy třídy jsou: objekt zajišťující komunikaci s MySQL serverem (datový typ MYSQL – součást připojené knihovny, viz kapitola 4.2.1), informace zdali se povedlo s databázovým systémem spojit (bool) a výpis chybových hlášení (pouze pro ladění, wxString).

V této třídě jsou vytvořeny dvě datové struktury. První se nazývá **testVratStruct** a využívá se k vypsání všech testů z databáze. Obsahuje čtyři atributy: ID testu (int), datum vygenerování, datum použití a poznámku k testu (všechny wxString). Druhá se jmenuje **testPorovnStruct** a slouží z výpisu podobnosti testů. Obsahuje dva atributy a to informace o testu (testVratStruct) a procentuálně vyjádřená podobnost (float). Pro obě tyto struktury jsou definovány prototypy dynamických polí (5.1.4).

V **konstruktoru** této třídy, jehož vstupem je jméno a heslo (wxString), je navázáno spojení s databázovým serverem MySQL. Po úspěšném spojení je serveru oznámena znaková sada klienta (v tomto případě cp1250, tedy znaková sada systému Windows). Pokud se vše zdaří, nastaví se proměnná indikující úspěšné spojení. Po vytvoření objektu by měla být ihned zavolána funkce **TestMySQL**. Jejím jediným úkolem je postupně spustit tři privátní funkce a vrátit informaci jestli všechny proběhly úspěšně (bool).

1. Funkce **vytvorDatabazi** – pokusí se vytvořit databázi s názvem „DBppgt“, se znakovou sadou utf8 a českými národními znaky (utf8_czech_ci) a pokud již taková databáze existuje, nechá ji beze změn.
2. Funkce **vytvorTabulky** – pokusí se vytvořit tři tabulky, které program potřebuje pro svoji funkci (viz Obrázek 5.2: Schéma použité databáze), pokud některá z těchto tabulek již existuje, nechá ji program beze změn.

3. Funkce **testSelectInsertUpdate** – ověří, jestli dostačují oprávnění, která má účet zadáný uživatelem. Pomocí tabulky s názvem „Pomocna“ se pokusí vykonat SQL příkazy: Insert, Select, Update a Truncate table (Delete).



Obrázek 5.2: Schéma použité databáze

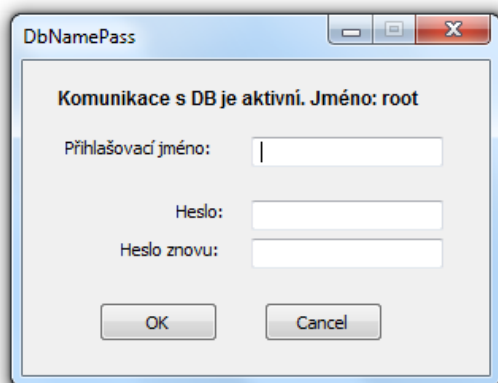
Funkce s názvem **vlozitTest** má na vstupu pole objektů „popisProGenOtazky“ (kapitola 5.4.1). Do tabulky s názvem „Testy“ vloží aktuální čas (datumVygenerovani) a poté sestrojí SQL dotaz (Select) pro zjištění id vloženého testu (idTestu). Pole idTestu má nastaveno autoincrement, aby nebylo nutné někde ukládat hodnotu id naposledy vloženého testu. S tímto získaným id vloží všechny otázky a odpovědi testu do databáze pod příslušnými čísly a v příslušném formátu (do tabulky „OtazkyOdpovedi“). Funkce **porovnatTest** má na vstupu pole objektů s otázkami a id aktuálního testu (int). Pro každou odpověď (v potaz je brán i text otázky) vytvoří SQL dotaz, který do tabulky „Pomocna“ vloží id všech testů, které ji obsahují (vynechán je aktuální test, tedy test s id které je na vstupu). Poté je vygenerován SQL dotaz, který do stejné tabulky vloží i id testů obsahující danou otázku (opět kromě aktuálního). Během tohoto vkládání je počítán počet otázek plus počet odpovědí. Nakonec je vytvořen SQL dotaz (Select), který vybere maximálně pět procentuálně nejpodobnějších testů (musí mít podobnost větší jak 0 %). Výsledek je uložen do pole struktury typu „testPorovnStruct“ a tabulka s názvem „Pomocna“ je vyprázdněna (SQL příkaz Truncate Table).

Funkce **pridejZaznamPouziti** má na vstupu id aktuálního testu (int) a poznámku, která bude k testu přidána (wxString). Tato funkce pouze vytvoří SQL dotaz (Update)

a k testu s požadovaným id vloží aktuální datum a tuto poznámku. Funkce *vypisTesty* po jejímž spuštění je vytvořen SQL dotaz (Select), který vypíše všechny informace o všech testech. Výsledek uloží do pole struktur typu „testVratStruct“. Funkce *nactiTest* má na vstupu id požadovaného testu (int). V těle této funkce je vytvořen SQL dotaz (Select), který z tabulky „OtazkyOdpovedi“ vrátí otázky a odpovědi požadovaného testu. Výsledek je uložen do pole objektů typu „popisProGenOtazky“ (kap. 5.4.1). Funkce *vymazTest* má na vstupu id testu (int) a jejím jediným úkolem je vymazat z obou tabulek záznamy s tímto „idTestu“.

5.3.2 Dialogové okno „DbNamePassDlg“

Dialogové okno umožňuje zadání jména a hesla uživatelského účtu, pomocí kterého se lze připojit k databázovému systému. Opět je vytvořeno pomocí designéru prostředí wxDev-Cpp. V *konstruktoru* jsou vytvořeny všechny grafické prvky a pomocí jednoho z prvků wxStaticText je vypsán popis informující o aktuálně zadaném účtu. Funkcionalita okna je zajištěna dvěma funkcemi, jejichž výstupem jsou zmiňované údaje (*GetName* a *GetPassword*), a reakcí na stisk tlačítka OK se zkontroluje, zda je zadáno uživatelské jméno a zda jsou stejná hesla v obou prvcích wxTextCtrl.



Obrázek 5.3: Dialogové okno „DbNamePassDlg“

5.4 Vygenerování testu

5.4.1 Třída „popisProGenOtazky“

Tato třída zajišťuje uchování parametrů otázky (předané dialogovým oknem dlgVyberOtazek) nebo již konkrétní otázku a možné odpovědi (po vygenerování testu nebo jeho načtení z databáze). Třída obsahuje osm atributů, třináct metod a čtyři statické funkce. Je zde definován prototyp dynamického pole (5.1.4). Atributy této třídy jsou: požadované parametry otázky (wxString, třikrát int), otázka (wxString), odpovědi

(wxArrayString), seznam pro generování použitelných otázek ze souboru (wxArrayInt) a číslo již vybrané otázky ze souboru (int).

Třída obsahuje tři GET funkce pro setřídování: **GetPocetMoznych**, jež má na výstupu počet otázek ze souboru, které odpovídají parametrům požadované otázky v objektu, **GetCisloOtazky**, která má na výstupu číslo vybrané otázky ze souboru a **GetCestaSouboruProPorovnani**, která má na výstupu cestu k souboru s otázkami.

Třída má **dva konstruktory**: jeden má na vstupu pouze otázku a odpověď (pro načítání testů z databáze) a druhý má na vstupu parametry testu, jehož jedinou úlohou (mimo uložení parametrů) je vytvoření entropie pro funkci „rand()“ generující náhodná čísla. To se provede příkazem: `srand(vložená_entropie)`

Statická funkce **ExistujeTest** ověřuje, zdali v dynamickém poli otázek, které má na vstupu, existuje test. Test musí mít více jak pět otázek, žádná otázka nesmí být bez textu a musí obsahovat odpovědi. Statická funkce **VygenerujObsahOtazek** má na vstupu dynamické pole otázek (tedy objektů této třídy). Pole je nejdříve setříděno podle cest k souborům jednotlivých otázek. To znamená, budou-li v poli dvě otázky, které by měli být vygenerovány ze stejného souboru, budou bezprostředně za sebou. Tohoto funkce využívá a sestavuje menší dynamické pole otázek se stejnou cestou k souboru otázek a to poté předá funkci **VygenerujOtazky**, která otázky vygeneruje. Po průchodu všemi otázkami je pole náhodně přeházeno.

Statická funkce **VygenerujOtazky** má na vstupu dynamické pole otázek se stejným souborem ze kterého mají být vygenerovány. Nejdříve ověří existenci souboru v souborovém systému a poté začne soubor číst pomocí tříd `wxFileInputStream` a `wxTextInputStream`. Nejdříve je přeskočen název tématu a popis souboru a ověřuje se, jestli nebylo dosaženo konce tohoto souboru. Poté následují otázky a u těch se v první fázi počítá pouze počet správných a špatných odpovědí. Tyto údaje jsou předány proceduře **TestOtazky**, která je zavolána pro každou otázku testu. Procedura **TestOtazky** ověří, jestli může být otázka v testu vytvořena pomocí aktuální otázky v souboru (jestli je dostatek správných i špatných odpovědí) a pokud ano je číslo otázky přidáno do pole otázek použitelných při generování.

Dalším krokem je seřazení dynamického pole otázek testu, podle počtu otázek v souboru, kterými může být zastoupena. Pokud má první otázka v poli tuto hodnotu nulovou je funkce ukončena. Následuje zavolání funkce **VygenerujKonkOtazku** pro

každou otázku testu. Tato funkce vybere z pole možných otázek v souboru jednu konkrétní (pomocí statické funkce *NahodneCislo*) a tu uloží. Po výběru čísla otázky ze souboru je pro všechny ostatní otázky testu zavolána procedura *SmazMoznost* a ta odstraní toto číslo z pole možných otázek ze souboru. Poté je soubor otázek čten opět od začátku a jednotlivé otázky ze souboru (pole wxArrayString s otázkou a všemi odpověďmi) jsou předány příslušným objektům otázek v testu, respektive proceduře této otázky (DotvorOtazku).

Procedura *DotvorOtazku* pole řetězců wxArrayString rozdělí na otázku, správné odpovědi (wxArrayString) a špatné odpovědi (wxArrayString). Poté je pomocí funkce generující náhodná čísla určen přesný počet správných i špatných odpovědí. Náhodně se z polí wxArrayString vymažou přebytečné odpovědi a tato pole se sloučí – získá se pole všech odpovědí a to je náhodně promícháno tak, aby se vždy vybrala jiná množina odpovědí s jiným pořadím.

5.4.2 Třída „MyThread“

Tato třída slouží k vygenerování a porovnávání testu. Je odvozena od třídy wxThread, pomocí které se vytvářejí vícevláknové aplikace. Obsahuje pět atributů a dvě metody. Atributy jsou jméno a heslo pro připojení k databázi (wxString), pole otázek („popisProGenOtazky“), id testu (int) a okno („PpGTfrm“), kterému se předá zpráva o dokončení běhu. V této třídě je definován nový typ eventu pro předání informace o ukončení vlákna (5.1.2).

Konstruktor má pouze tu funkci, že uloží všechny vstupní parametry do atributů objektu. Funkce *Entry* je poděděna od objektu wxThread a do ní je umístěna operace, kterou má vlákno vykonat. V tomto případě mohou být dva postupy vykonávání. Pokud je id vstupního testu rovno „-1“ jsou vyvolány dvě metody VygenerujObsahOtazek (5.4.1) pro vygenerování testu a vlozitTest (5.3.1) pro vložení testu do databáze. Poté je vlákno na jednu sekundu uspáno právě kvůli vkládání testů do databáze (aby v jednu sekundu nebylo možné vygenerovat dva testy). Následuje volání funkce porovnatTest (5.3.1) a zformátování řetězce s výpisem nejpodobnějších testů.

Pokud id vstupního testu bude různé od hodnoty „-1“, dojde pouze k porovnání testu pomocí funkce porovnatTest a zformátování řetězce s výpisem nejpodobnějších testů. Poté je hlavní okno informováno o ukončení vlákna a je mu předán jeho výstup: vstupní i výstupní id testu a řetězec s podobnostmi testů (vše pomocí eventu 5.1.2)

5.5 Nástroje pro vytvoření tiskové sestavy

5.5.1 Třída „MyPanel“

Pomocí této třídy je vytvořen „papír“, na kterém se v programu zobrazují texty. Obsahuje sedm atributů a šestnáct metod. V této třídě je vytvořena pomocná struktura s názvem *zaznamNaPapire* a také je zde definován prototyp dynamického pole těchto struktur (5.1.4). Struktura obsahuje proměnné, které identifikují text v okně: řetězec (wxString), barvu (wxColour), písmo (wxFont) a souřadnici (wxPoint). Atributy třídy jsou: vybrané okno, v němž se panel nachází (PpGTfrm), jednotlivé sloupce testu (wxArrayString), x a y souřadnice sloupců testu (2× wxArrayInt), písmo testu (wxFont), barva testu (wxColour) a pole záznamů (dynamické pole struktur zaznamNaPapire).

Tato třída je potomek tří tříd: wxPanel (grafický prvek, do něhož lze vkládat další prvky), wxMouseEvent (třída s eventy, které lze vyvolat myší) a wxEraseEvent (třída obsahující event vyvolaný při překreslování prvku, např. při změně velikosti okna). Toto umožňuje tomuto prvku vytvořit tabulku eventů (5.1.2).

Konstruktor této třídy pouze uloží referenci na hlavní okno a nastaví barvu pozadí. Procedura *pridej* pomocí čtyř vstupních parametrů přidá na panel libovolný text a zobrazí aktuální podobu panelu. Procedura *zmen* umožňuje u libovolného textu změnit obsah, barvu či písmo. Procedura *odeber* má na vstupu číslo textu, který má odstranit a pokud text existuje, bude z panelu vymazán. Procedura *odeberVse* vymaže všechny texty, které se na panelu nacházejí. Funkce *dejZaznam* má na vstupu číslo (uint) označující číslo záznamu a vrátí požadovaný záznam. Funkce *dejVsechny* vrátí všechny texty (záznamy) uložené na panelu. Funkce *dejPocet* vrátí počet záznamů na panelu (uint).

Metody, které byly do této doby zmíněny, sloužily pro správu textů (popisku, nadpisu a dalších řetězců) na panelu. Další zmíněné metody budou sloužit pro úpravu vloženého testu. Procedura *pridejTest* uloží test (s vybraným stylem a barvou písma) do panelu s tím, že souřadnice všech sloupců jsou 0, 0 (levý horní roh). Procedura *pridejTestNaMisto* také vloží test na panel (s požadovanou barvou a písmem), ale souřadnice prvního sloupce je zadána (wxPoint) a souřadnice dalších sloupců se vypočítá pomocí proměnné určující rozestup (int). Procedura *odeberTest* odebere všechny sloupce testu z panelu.

Všechny tyto metody na konci svého úkolu volají proceduru **ZobrazPridane** s tím, že jako vstupní parametr jí dají nástroj wxClientDC (viz 5.1.1) a označí, že chtějí učitelskou verzi testu (vyplněnou). Tato procedura tedy nejdříve úplně vymaže obsah panelu a poté do něho vykreslí všechny texty (metoda drawText) i test (procedura vykresleniSloupce). Procedura **vykresleniSloupce** zajišťuje vykreslení požadovaného sloupce testu (int – pořadí v poli testů). V první fázi vypočítá rozestupy mezi řetězci (v závislosti na velikosti písma) a poté pod sebe vypisuje jednotlivé řetězce. Pokud se jedná o otázku, je vypsána běžným způsobem a pokud se jedná o odpověď je před ní vykreslen čtverec o dané velikosti (dle velikosti písma). Pokud je požadována vyplněná verze testu je vykreslena ještě značka u správné odpovědi (pro menší písma je čtverec vyplněn černě a pro větší je vykreslena značka – drawCheckMark).

Prvnímu ze zděděných eventů „background erase event“ je přiřazena procedura **MyPanelEraseBackground** a ta při překreslování okna vyvolá metodu ZobrazPridane (překreslení všech textů i sloupců testu). Event reagující na kliknutí levého tlačítka myši na plochu panelu má přiřazenu proceduru **MyPanelLeftDown**. Ta z hlavního okna získá číslo aktuálně vybraného textu a přemístí ho na místo, kam bylo kliknuto. Pokud je číslo aktuálně vybraného textu -1 (nevybrán žádný prvek) a zároveň je v hlavním okně vyplněn nějaký text, je na panelu zobrazen (nikoliv přidán, takže při prvním překreslení se již neobjeví). Pokud je číslo prvku -1 a žádný text vyplněn není, není provedena žádná akce. Event pravého tlačítka má přiřazenu reakci **MyPanelRightDown**. Tato má podobnou funkci jako reakce na levé tlačítko s tím rozdílem, že přemísťuje aktuálně vybraný sloupec testu.

5.5.2 Třída „MyRower“

Při vkládání na „papír“ je celý test rozdělen do sloupců a tento nástroj slouží k základnímu formátování tohoto textu. Tato třída obsahuje jeden atribut a tři metody. Jediným atributem je pole otázek (viz kapitola 5.4.1). První z metod je **konstruktor** třídy, který uloží referenci na pole otázek, které se mají zformátovat.

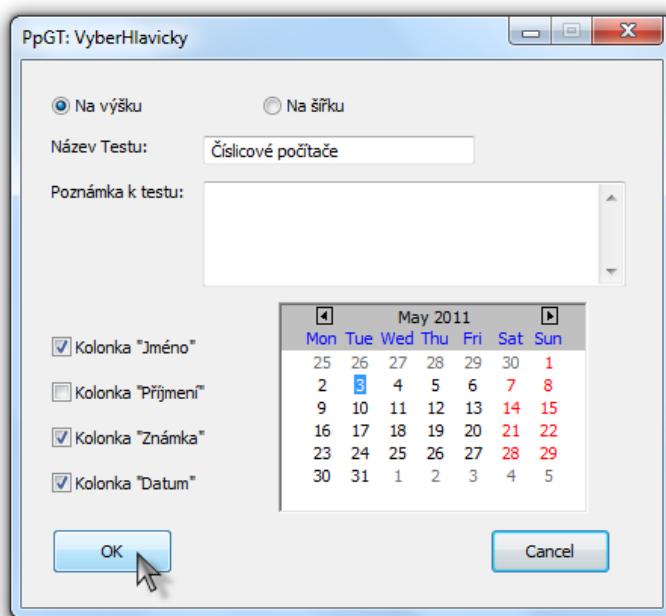
Jediná z vnějšku třídy volatelná funkce je funkce **MyRow** na jejímž výstupu je pole řetězců, kde každý řetězec představuje jeden sloupec testu. Vstupem jsou dvě čísla představující počet sloupců a maximální počet písmen ve sloupci (dvakrát int). Funkce MyRow zajišťuje rovnoměrné rozložení otázek ve sloupcích tím způsobem, že mezi vložením jednotlivých otázek, sloupce vždy seřadí od nejmenšího po největší. Tím je

zaručeno, že se další otázka vloží do prozatím nejkratšího sloupce. Nakonec je pole sloupců seřazeno opačně (od největšího), aby byl první sloupec testu nejdelší.

Metoda **RowOtazka** má na vstupu číslo vkládané otázky a také maximální délku řádku sloupce. V prvním kroku je text otázky rozdělen na jednotlivá slova pomocí třídy `wxStringTokenizer` s parametry `wxDEFAULT_DELIMITERS` (rozdělení podle bílých znaků – mezera, tabulátor ...) s tím, že jsou ignorována prázdná slova (například pokud jsou v textu dvě mezery za sebou – parametr `wxTOKEN_STRTOK`). Potom se jednotlivá slova vkládají do sloupce tak, aby nemusela být dělena. Jednopísmenná slova v otázce se automaticky spojí s následujícím slovem, aby nedošlo k osamocení písmenu na konci řádku. Pokud je některé ze slov delší, než maximální počet písmen v řádku, je rozděleno podle délky řádku a vloženo na následující prázdné řádky. Otázky jsou zformátovány stejným způsobem s tím rozdílem, že na začátek každého řádku je vloženo osm mezer (případně pět mezer a znaky „[*]“ či „[]“ pokud se jedná o začátek otázky)

5.5.3 Dialogové okno „VyberHlavickyDlg“

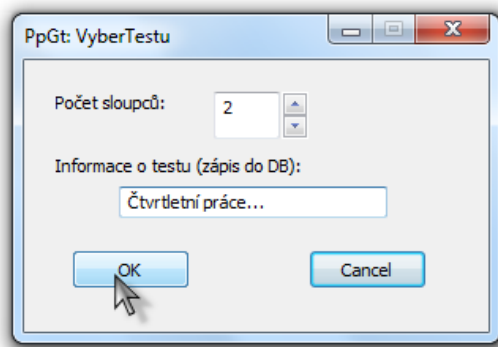
Toto dialogové okno (Obrázek 5.4) slouží k zadání informací, které se mají vyplnit jako text na tiskovou podobu testu. Je vytvořeno pomocí designéru prostředí `wxDev-C++`. Jeho jedinou funkčnost zajišťuje metoda **buttonOKClick**, která reaguje na stisk tlačítka OK a pouze ověří, zdali uživatel vyplnil název testu (jediná povinná součást). Okno je doplněno o GET funkce k získání všech parametrů.



Obrázek 5.4: Dialogové okno pro vybrání hlavičky testu

5.5.4 Dialogové okno „VyberTestuDlg“

Dialogové okno slouží pro výběr rozestavení testu na „papír“ a je vytvořeno pomocí designéru prostředí wxDev-C++. Je doplněno pouze o tři metody, kde jedna nastavuje maximální a minimální hodnotu v prvku wxSpinCtrl, ve kterém se nastavuje počet sloupců pro rozložení testu, další dvě slouží k získání údajů z okna.



Obrázek 5.5: Dialogové okno pro výběr vzhledu testu

5.5.5 Třída „MyPrintout“

Systém tisku vyžaduje vytvoření vlastní třídy odvozené od třídy wxPrintout, která bude vytvářet grafickou podobu tisknuté stránky. Třída má dva atributy, dvě vlastní a tři přetížené metody od objektu wxPrintout. Atributy jsou: číslo představující počet kopií k tisku (int) a panel, jehož obsah bude tisknut (Třída „MyPanel“).

Konstruktor pouze převezme tyto dva atributy a navíc také titulek tištěného dokumentu, který je předán předkovi. Odvozená metoda **GetPageInfo** vrací informaci o počtu stránek, které lze vytisknout, případně zvolit k tisku. Zde je minimum jeden dokument a maximum je určeno počtem požadovaných kopií. Další odvozená funkce **HasPage** má na vstupu číslo stránky a musí vracet hodnotu true (datový typ bool) pro všechny strany které mohou být vytištěny.

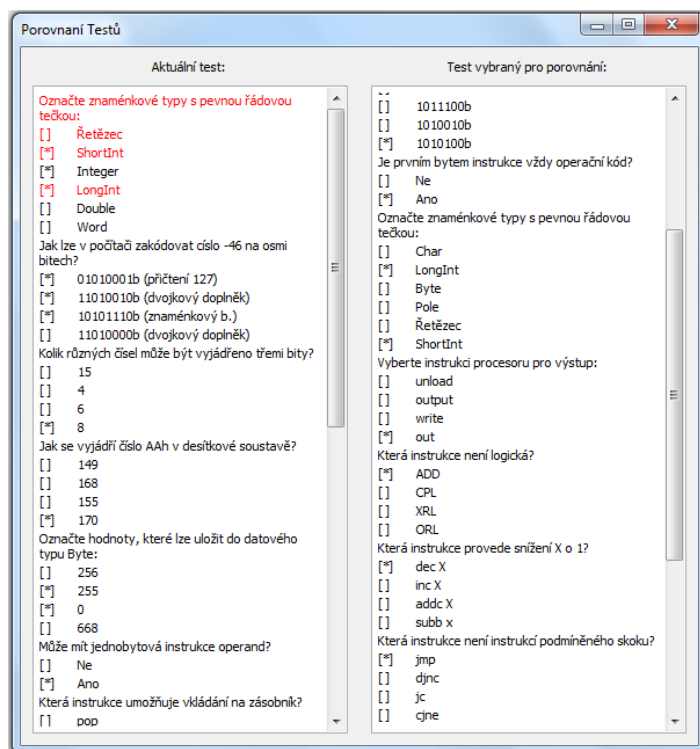
Poslední odvozená funkce **OnPrintPage** má na vstupu také číslo stránky. Funkce musí vytvořit podobu stránky, která se má vytisknout a vrátit informaci, zda se vytvoření zdařilo (bool). U první tištěné stránky je nastaven učitelský mód (vytiskne vyplněný test) a další stránky testu jsou vytištěny nevyplněné. Stránka se vytvoří pomocí metody **DrawPage**. V té se nejdříve nastaví poměr velikostí tištěné strany a plátna, na které se stránka vykresluje. Lze například docílit toho, že se vykresluje na plátno menší velikosti (např. 100×500 pixelů) a to se poté roztáhne na jeden rozměr stránky (zachovává se poměr stran plátna, takže se obrázek roztáhne tak, aby jeden

rozměr stránky zcela zaplnil). Poté se plátno umístí na střed druhého rozměru stránky. V této aplikaci je velikost plátna zvolena jako násobek rozměru stránky A4, takže se vždy roztáhne na oba rozměry tištěné stránky. Výjimku tvoří možnost návrhu stránky na šířku a následné tisknutí na výšku. V tom případě by byla stránka vykreslena na šířku a umístěna do středu stránky ve zmenšené podobě. Vykreslení stránky zajišťuje metoda `ZobrazPridane (MyPanel)`, které je předán nástroj `wxDC (5.1.1)` pro tisk (`wxPrinterDC`).

5.6 Doplnující funkce

5.6.1 Dialogové okno „Porovnanitestudlg“

Toto dialogové okno slouží k vizuálnímu porovnání aktuálního testu s vybraným testem z databáze. Opět je vytvořeno pomocí designéru `wxDev-C++` a doplněno o tři metody. Jeho funkcionalitu zajišťuje metoda ***VypisTesty***, která nejdříve kompletně vypíše test vybraný z databáze. Poté vypisuje aktuální test. Otázku aktuálního testu se pokusí nalézt v testu z databáze (funkce ***VyhledejOtazku***) a pokud uspěje, vypíše se otázka červeně, pokud neuspěje tak černě. Nalezená otázka je uchována a každá vkládaná odpověď je testována, jestli se v ní nachází (funkce ***VyhledejOdpoved***). Pokud ano vypíše se také červeně, pokud ne je vypsána černě. V případě, že se otázka v testu z databáze nenachází, jsou odpovědi vypsány černě.

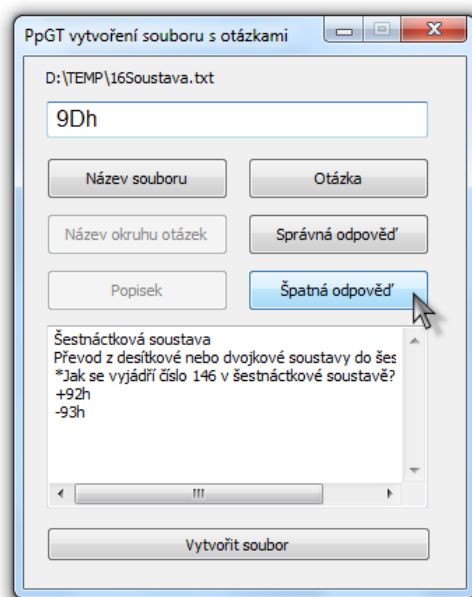


Obrázek 5.6: Ukázka porovnání testů

5.6.2 Dialogové okno „vytvorSouborTestuDlg“

Toto dialogové okno slouží k vytvoření souboru otázek v požadovaném formátu (jedná se o velmi jednoduchý editor). Okno je vytvořeno pomocí designéru prostředí wxDev-C++ a jeho funkcionalitu zajišťují metody reagující na stisk tlačítek.

Při stisku tlačítka „Název Souboru“ je vyvolán dialog pro výběr souboru (wxFileDialog) a jméno souboru je uloženo do popisku (wxStaticText). Toto tlačítko je možné stisknout kdykoliv a jméno souboru změnit. Ostatní tlačítka jsou blokována a povolována v závislosti na struktuře souboru (např. „Název okruhu otázek“ lze změnit pouze v úvodu). Všechny zadávané údaje jsou jednořádkové, výjimkou je popis, který může být víceřádkový a jeho zadávání se ukončí zadáním prázdného řádku. Při stisknutí tlačítka „Vytvořit soubor“ je ověřeno, zda byl zadán název souboru a pokud ano, je pomocí nástroje wxFile soubor vytvořen a jsou do něho zapsány požadované řádky.



Obrázek 5.7: Jednoduchý editor souboru otázek

5.7 Hlavní okno aplikace „PpGTfrm“

Hlavní okno celé aplikace, které využívá výše zmíněné moduly pro chod celé aplikace. Je vytvořeno pomocí designéru prostředí wxDev-C++ a doplněno o panel pro vykreslování testu (5.5.1). Funkcionalita je mimo eventů zajištěna deseti atributy a třinácti metodami. Atributy slouží k uchování informací, jako jsou: ID testu, který je vypsán na „papíře“, ukazatel na vlákno aplikace, otázky aktuálně vygenerovaného či

načteného testu, výchozí písmo a barvu textu, informace o dostupnosti databáze a přístupové údaje, název aktuálně načteného souboru s hlavičkou či indikátor změn.

V **konstruktoru** okna dochází k vytvoření grafických prvků okna, nastavení výchozích hodnot atributů a ověření dostupnosti databáze. Ověření dostupnosti databáze se provádí pomocí souboru, ve kterém je uloženo jméno a heslo, které uživatel již dříve zadal. Tento soubor („db.dat“) se nachází ve složce „conf“, která je na stejném místě, jako spustitelný soubor tohoto programu. Pokud se pomocí těchto údajů nepovede s databází spojit, jsou automaticky zablokovány všechny funkce, které ji využívají. Zde je poté nutné zvolit v menu položku „Spojení s databází“ a pomocí zobrazeného dialogového okna zadat potřebné údaje. Ihned poté jsou údaje uloženy do zmíněného souboru a ověřena dostupnost databáze. Pokud se spojení zdaří, jsou funkce využívající databázi zpřístupněny (funkcí *nastavGUIpodleDB*).

Okno má vytvořeno standardní ovládací systém z položek „Nový“, „Načíst“, „Uložit“, „Uložit jako“ nebo „Konec“. Při použití těchto možností je kontrolováno, zda uživatel požadovanou akci nepřichází o neuložená data a případně vypsáno varování či dotaz na uložení. Položky „Načíst“, „Uložit“ a „Uložit jako“ ukládají pouze hlavičku testu a to umožňuje její vícenásobné použití. K uložení hlavičky jsou využity nástroje wxFile, wxFileOutputStream a wxDataOutputStream, které nejdříve zapíší řetězec „PpGTulozenaHlavicka“ a poté veškerá data hlavičky: rozložení papíru na výšku / na šířku a u textů na papíře (metoda *dejVsechny* třídy *MyPanel*) jsou zapsány všechny dostupné údaje, jako jsou podtržení, pozice, styl, barva písma atd. Při načítání je soubor otevřen pomocí nástrojů wxFile, wxFileInputStream a wxDataInputStream, ověří se přítomnost řetězce „PpGTulozenaHlavicka“ a pokud se v souboru nachází, jsou načteny všechny zmíněné informace o hlavičce. Položka „Nový“ vymazává veškeré nastavené hodnoty (obnoví výchozí nastavení).

Pokud uživatel v menu vybere možnost „Vytvořit nový test“, je mu zobrazeno dialogové okno pro výběr nastavení testu (Dialogové okno „dlgVyberOtazek“). Pokud uživatel okno potvrdí je vymazán aktuální test (metoda *vymazTest*) a vytvořeno vlákno, které test vygeneruje, uloží do databáze a porovná s ostatními testy v databázi (5.4.2). Vlákno po svém dokončení vytvoří event, na který je reagováno procedurou **OnThread**. Ta z tohoto eventu přebere veškeré dostupné informace (jestli byl test vygenerován, ID testu v databázi, řetězec popisující nejpodobnější testy) a poté je zobrazí nebo uživatele informuje prostřednictvím chybového hlášení. Pokud je test úspěšně vygenerován

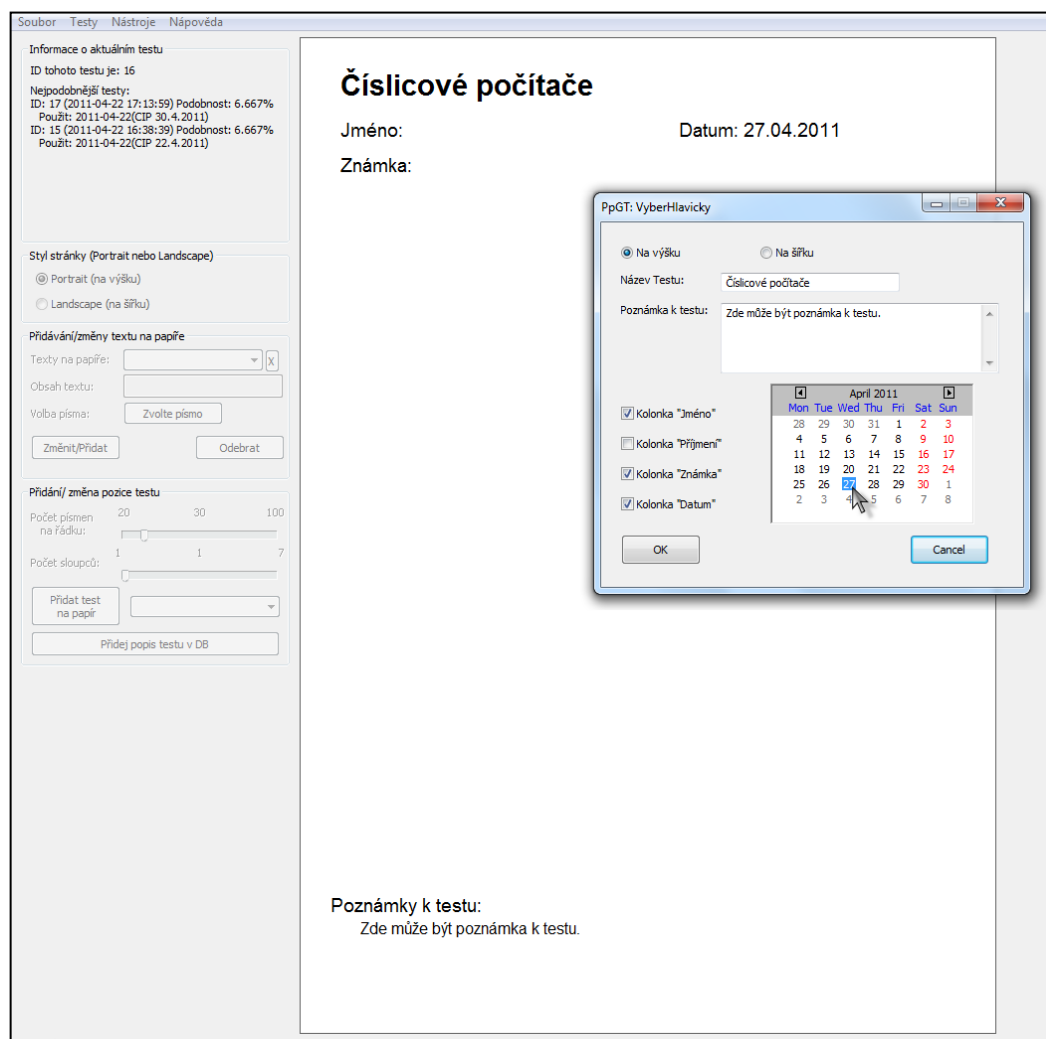
povolí se také menu pro generování testu podle stejné šablony. Pokud uživatel zvolí tuto možnost, je vlákno spuštěno znovu a vytvořen nový test (včetně porovnání atd.). Test je také možné načíst z databáze, pokud tuto možnost uživatel zvolí, je mu zobrazen pomocí dialogového okna seznam dostupných testů (metoda *nactiTest*) a vybraný test je z databáze načten pomocí metody *nacistTest* z třídy *praceMySQL* (5.3.1). Poté je opět spuštěno zmíněné vlákno, které ovšem tentokrát test pouze porovná a vypíše výsledky.

Pokud uživatel vybere možnost „Porovnat s testem v DB“, je ověřena existence aktuálního testu. Pokud opravdu existuje, je opět dán uživateli seznam dostupných testů (*nactiTest*) a oba jsou zobrazeny v okně „PorovnanitestuDlg“ (5.6.1). Při zvolení možnosti „Vymazat test z DB“, je uživateli poskytnut seznam testů v databázi a po výběru konkrétního testu je vymazán pomocí metody *vymazTest* třídy *praceMySQL*. Pokud uživatel vybere možnost „Nový soubor otázek“, je pouze aktivováno dialogové okno *vytvorSouborTestuDlg* (5.6.2). Zobrazení položky „O programu“ je vytvořeno pomocí prvku *wxMessageDialog* knihovny *wxWidgets* a tímto nástrojem jsou také vytvořena všechna chybová hlášení.

Při výběru manuálního vložení hlavičky, je povoleno menu na levé straně programu. Zde je možné vybírat rozložení panelu představující papír a vkládat na něj libovolný text. Při změně rozložení panelu jsou změněny rozměry jak toho panelu, tak prvku *wxScrolledWindow*, který zajišťuje zobrazení posuvných nástrojů na pravé straně či vespod okna tak, aby bylo možné program obsluhovat i nemaximalizovaný. Pokud je vybrána možnost „Zvolte písmo“ zobrazí se panel *wxFontDialog*, který představuje obvyklé okno pro výběr písma realizované knihovnou *wxWidgets*. Při stisknutí tlačítka „Změnit/Přidat“ je ověřeno, jestli je vyplněn nějaký text v grafickém prvku *wxTextCtrl*. V případě že ano, jsou přečteny všechny údaje (řetězec, styl a barva písma) a text je přidán na panel. Pokud je ovšem v prvku *wxChoice* zvolen již existující text dojde k jeho změně dle těchto údajů. Tlačítko „Odeber“ ověří, zdali je vybrán existující text na panelu a pokud ano, vymaže jej. Tyto události jsou realizovány pomocí funkcí samotného panelu *MyPanel* (5.5.1). Tlačítko „X“ slouží k odstranění vybraného textu z prvku *wxChoice* tak, aby uživatel mohl vložit nový text.

Při volbě automatického vložení hlavičky je zobrazeno dialogové okno pro výběr obsahu hlavičky – *VyberHlavickyDlg* (5.5.3). Po jeho potvrzení jsou z panelu odebrány všechny texty a texty vybrané v dialogovém okně se vloží na definované

pozice (pomocí metody přidej třídy MyPanel) v závislosti na tom jestli byl vybrán test na výšku či na šířku.



Obrázek 5.8: Hlavní okno aplikace při výběru parametrů hlavičky

Výběr manuálního vložení testu povolí spodní menu na levé straně obrazovky. Zde je možné zvolit počet sloupců a počet písmen v sloupci a stisknout tlačítko „Přidat test na papír“. Po jeho stisknutí se objeví dialogové okno pro výběr písma, poté je test přidán na panel v požadovaném formátu (metoda přidejTest třídy MyPanel) a je uloženo jeho ID. Toto je využito při stisknutí tlačítka „Přidej popis testu v DB“, kde se uživateli zobrazí okno pro zápis textu a tento text je přidán jako poznámka testu s tímto ID v databázi.

Při výběru automatického vložení testu je ověřena existence vygenerovaného či načteného testu a poté je uživateli dán výběr počtu sloupců testu a možnost zápisu poznámky do databáze. Po zvolení požadovaného počtu sloupců a textu je vymazán

případný dříve vložený test a aktuální test je vložen na předem definovaná místa s definovanými rozestupy. V případě výběru poznámky je tato poznámka zapsána do databáze (metoda `pridejZaznamPouziti` třídy `praceMySQL`).

Poslední důležitou součástí programu je možnost tisku, která je realizována třídou `wxPrinter`. Nejdříve se uživateli zobrazí okno pro nastavení tisku, poté je zaměněn údaj o počtu kopií s údajem o počtu tisknutých stran (aby se například při tisku dvou testů na stránku, vytiskly dva testy na jeden papír nikoli každý na nový papír). Tisk je realizován prostřednictvím třídy `MyPrintout` (5.5.5). Je zde také možnost náhledu, která je realizována prostředky `wxPrintPreview` a `wxPreviewFrame` (zobrazí se dvě strany testu, tedy jedna vyplněná a jedna prázdná).

6 Závěr

Zkoušení pomocí testů založených na výběru správných odpovědí z několika možností, je velmi často využívaná metoda pro ověřování vědomostí studentů. Bohužel je opakované vytváření takovýchto zkouškových testů časově velmi náročný problém, který si ovšem v dnešní době může vyučující usnadnit použitím některého programu specializujícího se na generování a tisk testů.

Na internetu se pro tento úkol vyskytuje velké množství programů. Nicméně ve velké míře se jedná o programy zaměřené na jednotlivé tematické okruhy a přidávání otázek je u nich velmi problematické, často i nemožné. Samozřejmě existují i programy specializované na vytváření zkouškových testů, na které je zaměřena první část této práce. Některé programy jsou volně dostupné jako například Generátor jedinečných zadání, QuizFaber nebo Test. Tyto programy ovšem nabízejí menší množství funkcí a tisk testů mnohdy není jejich nejsilnější stránkou. Lepší je to u programů jako jsou Test Generator, Quiz Builder nebo DoTest, ale tyto programy už ovšem nejsou volně dostupné a po uplynutí zkušební doby je nutné zakoupit licenci.

Ovšem ani tyto licencované programy někdy nesplní veškerá očekávání a proto bylo cílem druhé části této práce vytvořit co možná nejuniverzálnější řešení pro vytváření testů v předmětech ze základního kurzu bakalářského studia. Byla stanovena základní funkcionalita, kterou by měl program splňovat a vybrány vhodné nástroje k jejich realizování. Vývojovým prostředím bylo zvoleno prostředí wxDev-C++, což je prostředí jazyka C++ doplněné o nástroje knihovny wxWidgets. Pro uchovávání dat byl zvolen databázový systém MySQL.

Pomocí těchto nástrojů byl vytvořen program umožňující testy nejen generovat na základě uživatelského nastavení, ale i libovolně formátovat. Jeho součástí je také systém, který umožňuje uchovávat dříve vygenerované testy a ty poté porovnávat s testem aktuálně vytvořeným, aby pomohl vyučujícímu se rozhodnout, jestli test použít nebo raději vytvořit nový. Pro otestování vytvořeného řešení byl vybrán předmět „Číslicové počítače“, ze kterého byla vytvořena databáze otázek v požadovaném formátu a poté bylo pomocí programu vytvořeno několik verzí testů (příloha A a B).

Možností na rozšíření této aplikace je celá řada. Od možnosti zabezpečení otázek po možnost implementovat modul automatického testování, který by umožnil pomocí http serveru testovat studenty vzdáleně bez nutnosti vytváření a tisknutí testu.

Seznam použité literatury

- [1] SVOBODA, Luboš. *Osobní stránka - L. Svoboda* [online]. Version 11.5606. [Praha]: [Fakulta stavební (ČVUT)], 2001-07-30, datum poslední úpravy této stránky: 5. 8. 2010 [cit. 2010-10-01]. Generátor jedinečných zadání - Podrobný popis programu. Dostupné z WWW: <<http://people.fsv.cvut.cz/~svobodan/gejza/webgejza.htm>>.
- [2] GALLI, Luca. *Luca Galli's Site* [online]. [Turin (Italy)]: c2010 [cit. 2010-10-04]. Citované stránky: QuizFaber – Description, Main menu – About me. Dostupné z WWW: <<http://www.lucagalli.net/en/>>.
- [3] Fain & Company. *Test generator* [online]. [Atlanta]: c2010 [cit. 2010-10-10]. Citované stránky: Support – Update history, About us – Contact, Purchase. Dostupné z WWW: <<http://www.testshop.com/>>.
- [4] ŽOČEK, Petr. *Software - Petr Žoček ml.* [online]. [Hradec Králové]: 2002, Poslední aktualizace 28.listopadu 2004 [cit. 2010-10-13]. Test. Dostupné z WWW: <<http://software.zoczek.cz/?Cesta=./txt/test.php>>.
- [5] Tanida Inc. *Quiz Builder* [online]. [Toronto]: c2010, 3.1.2010 [cit. 2010-10-20]. Citované stránky: Contact, Download a free trial, How to buy, What's new. Dostupné z WWW: <<http://www.quiz-builder.com/>>.
- [6] Dosli. *Dosli – pomáháme vám učit* [online]. [Opava]: c1991 [cit. 2010-10-22]. Citované stránky: Ceny licencí, DoTest 4 – Úvod, O společnosti. Dostupné z WWW: <<http://web.dosli.cz/cze/index.html>>.
- [7] KATHIRESAN, Guru, et al. *WxDev-C++* [online]. [SourceForge], 2003 [cit. 2011-04-25]. Citované stránky: Developer Team, Welcome. Dostupné z WWW: <<http://wxdsgn.sourceforge.net/>>.
- [8] HINZ, Stefan, et al. *MySQL 5.5 Reference Manual* [online]. [USA]: [Oracle Corporation], c1997, Document generated on: 2011-04-22 [cit. 2011-04-25]. Citované stránky: History of MySQL, MySQL C API – Building Client Programs (+User Comments), MySQL Connector/C++, The Main Features of MySQL, What is MySQL?. Dostupné z WWW: <<http://dev.mysql.com/doc/refman/5.5/en/>>.

- [9] OLLIVIER, Kevin, et al. *WxWidgets: Cross-platform GUI Library* [online]. 2007 [cit. 2011-04-26]. Citované stránky: About – Features, About – Licence, About – History, Home. Dostupné z WWW: <<http://www.wxwidgets.org/>>.
- [10] SMART, Julian; HOCK, Kevin; CSOMOR, Stefan. *Cross-Platform GUI Programming with wxWidgets* [online]. First printing. Crawfordsville, Indiana (the United States): R.R. Donnelley & Sons, July 2005 [cit. 2011-05-02]. Drawing and Printing, s. 131-170. Dostupné z WWW: <http://ptgmedia.pearsoncmg.com/images/0131473816/downloads/0131473816_book.pdf>. ISBN 0-13-147381-6.
- [11] SMART, Julian, et al. *WxWidgets 2.8.12: A portable C++ and Python GUI toolkit* [online]. March, 2011 [cit. 2011-05-02]. Citované stránky: Event handling overview, wxArray. Dostupné z WWW: <http://docs.wxwidgets.org/2.8/wx_contents.html>.

Příloha A – Obsah CD

Bakalářská práce ve formátu PDF a XPS

Vytvořená databáze otázek z předmětu CIP

2x ukázka vygenerovaného testu z předmětu CIP

Jednotlivé moduly programu PpGT

Sestavený program PpGT

Spustitelný soubor PpGT.exe + potřebné soubory

Instalátor vytvořený pomocí programu Inno Setup 5

Ikona programu PpGT

Odkazy na použitý SW

Příloha B – Ukázka testu č. 1

Číslicové počítače

Jméno:

Datum: 27.04.2011

Známka:

Označte znaménkové typy s pevnou řádovou tečkou:

- ☐ Řetězec
- ☒ ShortInt
- ☒ Integer
- ☒ LongInt
- ☐ Double
- ☐ Word

Jak se vyjádří číslo AAh v desítkové soustavě?

- ☐ 149
- ☐ 168
- ☐ 155
- ☒ 170

Může mít jednobytová instrukce operand?

- ☐ Ne
- ☒ Ano

Která instrukce umožňuje vkládání na zásobník?

- ☐ pop
- ☐ call
- ☒ push
- ☐ apd

Na adrese 456 je uloženo číslo 2222 (dat typ Word), jak bude vypadat byte číslo 457?

- ☐ 00011100
- ☐ 00010000
- ☐ 00011000
- ☒ 00001000

V registru A je 01110011, vykoná se "anl A,X" (kde x = 01001101), co obsahuje nyní registr A?

- ☐ 00111110
- ☒ 01000001
- ☐ 11000001
- ☐ 01111111

Jak lze v počítači zakódovat číslo -46 na osmi bitech?

- ☒ 01010001b (přičtení 127)
- ☒ 11010010b (dvojkový doplněk)
- ☒ 10101110b (znaménkový b.)
- ☐ 11010000b (dvojkový doplněk)

Kolik různých čísel může být vyjádřeno třemi bity?

- ☐ 15
- ☐ 4
- ☐ 6
- ☒ 8

Označte hodnoty, které lze uložit do datového typu Byte:

- ☐ 256
- ☒ 255
- ☒ 0
- ☐ 668

V registru A je 01110011 a v C je 0. Co bude v registru A po instrukci "rl A"?

- ☐ 01101010
- ☐ 10111001
- ☐ 11100101
- ☒ 11100110

Jakou funkci provádí instrukce "mov X, Y":

- ☐ Odečte X od Y
- ☐ Vyměňuje obsahy Y a X
- ☒ Kopíruje Y do X
- ☐ Žádnou

Která instrukce provede sečtení registru X a A?

- ☒ add A, X
- ☐ dec A, X
- ☐ addc A, X
- ☐ subb A, x

Poznámky k testu:

Některé otázky mají více než jednu správnou odpověď...

Příloha C – Ukázka testu č. 2

Číslicové počítače		
Jméno:	Příjmení:	Datum: 22.04.2011
		Známka:
<p>Jakými způsoby lze kódovat záporná čísla ve dvojkové soustavě?</p> <p><input checked="" type="checkbox"/> Přičtení konstanty</p> <p><input type="checkbox"/> Trojkový doplněk</p> <p><input type="checkbox"/> Odečtení konstanty</p> <p><input type="checkbox"/> Bitový strom</p>	<p>Jak se vyjádří číslo 84 ve dvojkové soustavě?</p> <p><input type="checkbox"/> 0010101b</p> <p><input type="checkbox"/> 1011100b</p> <p><input type="checkbox"/> 1010010b</p> <p><input checked="" type="checkbox"/> 1010100b</p>	<p>Která instrukce není instrukcí rotace?</p> <p><input type="checkbox"/> rr</p> <p><input type="checkbox"/> rl</p> <p><input checked="" type="checkbox"/> clr</p> <p><input type="checkbox"/> rlc</p>
<p>Označte znaménkové typy s pevnou řádovou tečkou:</p> <p><input type="checkbox"/> Char</p> <p><input checked="" type="checkbox"/> LongInt</p> <p><input type="checkbox"/> Byte</p> <p><input type="checkbox"/> Pole</p> <p><input type="checkbox"/> Řetězec</p> <p><input checked="" type="checkbox"/> ShortInt</p>	<p>Vyberte instrukci procesoru pro výstup:</p> <p><input type="checkbox"/> unload</p> <p><input type="checkbox"/> output</p> <p><input type="checkbox"/> write</p> <p><input checked="" type="checkbox"/> out</p>	<p>Je prvním bytem instrukce vždy operační kód?</p> <p><input type="checkbox"/> Ne</p> <p><input checked="" type="checkbox"/> Ano</p>
<p>Která instrukce není instrukcí podmíněného skoku?</p> <p><input checked="" type="checkbox"/> jmp</p> <p><input type="checkbox"/> djnc</p> <p><input type="checkbox"/> jc</p> <p><input type="checkbox"/> cjne</p>	<p>Která instrukce provede snížení X o 1?</p> <p><input checked="" type="checkbox"/> dec X</p> <p><input type="checkbox"/> inc X</p> <p><input type="checkbox"/> addc X</p> <p><input type="checkbox"/> subb x</p>	<p>Která instrukce není logická?</p> <p><input checked="" type="checkbox"/> ADD</p> <p><input type="checkbox"/> CPL</p> <p><input type="checkbox"/> XRL</p> <p><input type="checkbox"/> ORL</p>
<p>Označte hodnoty, které v ASCII tabulce interpretují malá písmena:</p> <p><input checked="" type="checkbox"/> 120</p> <p><input type="checkbox"/> 46</p> <p><input type="checkbox"/> 64</p> <p><input type="checkbox"/> 90</p>	<p>Jak se vyjádří číslo BCh ve dvojkové soustavě?</p> <p><input type="checkbox"/> 10111101b</p> <p><input type="checkbox"/> 10111011b</p> <p><input type="checkbox"/> 10101101b</p> <p><input checked="" type="checkbox"/> 10111100b</p>	<p>Na kolik bitů se uloží datový typ Extended?</p> <p><input type="checkbox"/> 96</p> <p><input type="checkbox"/> 56</p> <p><input type="checkbox"/> 32</p> <p><input checked="" type="checkbox"/> 80</p>
<p>Poznámky k testu:</p> <p>Některé otázky mají více než jednu správnou odpověď (je nutné označit všechny).</p> <p>Podmínka pro splnění testu je 75% správně vyplněných otázek (9 otázek).</p>		

Číslicové počítače		
Jméno:	Příjmení:	Datum: 22.04.2011
		Známka:
<p>Jakými způsoby lze kódovat záporná čísla ve dvojkové soustavě?</p> <p><input type="checkbox"/> Přičtení konstanty</p> <p><input type="checkbox"/> Trojkový doplněk</p> <p><input type="checkbox"/> Odečtení konstanty</p> <p><input type="checkbox"/> Bitový strom</p>	<p>Jak se vyjádří číslo 84 ve dvojkové soustavě?</p> <p><input type="checkbox"/> 0010101b</p> <p><input type="checkbox"/> 1011100b</p> <p><input type="checkbox"/> 1010010b</p> <p><input type="checkbox"/> 1010100b</p>	<p>Která instrukce není instrukcí rotace?</p> <p><input type="checkbox"/> rr</p> <p><input type="checkbox"/> rl</p> <p><input type="checkbox"/> clr</p> <p><input type="checkbox"/> rlc</p>
<p>Označte znaménkové typy s pevnou řádovou tečkou:</p> <p><input type="checkbox"/> Char</p> <p><input type="checkbox"/> LongInt</p> <p><input type="checkbox"/> Byte</p> <p><input type="checkbox"/> Pole</p> <p><input type="checkbox"/> Řetězec</p> <p><input type="checkbox"/> ShortInt</p>	<p>Vyberte instrukci procesoru pro výstup:</p> <p><input type="checkbox"/> unload</p> <p><input type="checkbox"/> output</p> <p><input type="checkbox"/> write</p> <p><input type="checkbox"/> out</p>	<p>Je prvním bytem instrukce vždy operační kód?</p> <p><input type="checkbox"/> Ne</p> <p><input type="checkbox"/> Ano</p>
<p>Která instrukce není instrukcí podmíněného skoku?</p> <p><input type="checkbox"/> jmp</p> <p><input type="checkbox"/> djnc</p> <p><input type="checkbox"/> jc</p> <p><input type="checkbox"/> cjne</p>	<p>Která instrukce provede snížení X o 1?</p> <p><input type="checkbox"/> dec X</p> <p><input type="checkbox"/> inc X</p> <p><input type="checkbox"/> addc X</p> <p><input type="checkbox"/> subb x</p>	<p>Která instrukce není logická?</p> <p><input type="checkbox"/> ADD</p> <p><input type="checkbox"/> CPL</p> <p><input type="checkbox"/> XRL</p> <p><input type="checkbox"/> ORL</p>
<p>Označte hodnoty, které v ASCII tabulce interpretují malá písmena:</p> <p><input type="checkbox"/> 120</p> <p><input type="checkbox"/> 46</p> <p><input type="checkbox"/> 64</p> <p><input type="checkbox"/> 90</p>	<p>Jak se vyjádří číslo BCh ve dvojkové soustavě?</p> <p><input type="checkbox"/> 10111101b</p> <p><input type="checkbox"/> 10111011b</p> <p><input type="checkbox"/> 10101101b</p> <p><input type="checkbox"/> 10111100b</p>	<p>Na kolik bitů se uloží datový typ Extended?</p> <p><input type="checkbox"/> 96</p> <p><input type="checkbox"/> 56</p> <p><input type="checkbox"/> 32</p> <p><input type="checkbox"/> 80</p>
<p>Poznámky k testu:</p> <p>Některé otázky mají více než jednu správnou odpověď (je nutné označit všechny).</p> <p>Podmínka pro splnění testu je 75% správně vyplněných otázek (9 otázek).</p>		